

Novell ZENworks® for Servers

3.0.2

www.novell.com

ADMINISTRATION

March 11, 2005



Novell®

Legal Notices

Novell, Inc. makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

You may not export or re-export this product in violation of any applicable laws or regulations including, without limitation, U.S. export regulations or the laws of the country in which you reside.

Copyright © 1999-2004 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

U.S. Patent Nos. 5,910,803; 6,067,093. Patents Pending.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.

www.novell.com

Administration
[March 11, 2005](#)

Online Documentation: To access the online documentation for this and other Novell products, and to get updates, see www.novell.com/documentation.

Novell Trademarks

ConsoleOne is a registered trademark of Novell, Inc. in the United States and other countries.

eDirectory is a trademark of Novell, Inc.

GroupWise is a registered trademark of Novell, Inc. in the United States and other countries.

IPX is a trademark of Novell, Inc.

NCP is a trademark of Novell, Inc.

NDS is a registered trademark of Novell, Inc. in the United States and other countries.

NetExplorer is a trademark of Novell, Inc.

NetWare is a registered trademark of Novell, Inc. in the United States and other countries.

NetWare Core Protocol is a trademark of Novell, Inc.

NetWare Management Agent is a trademark of Novell, Inc.

NetWare SFT III is a trademark of Novell, Inc.

NLM is a trademark of Novell, Inc.

Novell is a registered trademark of Novell, Inc. in the United States and other countries.

Novell Client is a trademark of Novell, Inc.

Novell Technical Services is a service mark of Novell, Inc.

SPX is a trademark of Novell, Inc.

ZENworks is a registered trademark of Novell, Inc. in the United States and other countries.

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Contents

- About This Guide 21

- Part I Management and Monitoring Services**

- 1 Configuring Management and Monitoring Services 25**
 - Understanding Management and Monitoring Services 25
 - Management Site Services 25
 - Server Management 28
 - Traffic Analysis 28
 - ConsoleOne 29
 - Planning the Configuration 29
 - Defining Management Information Needs 30
 - Planning a Strategy to Manage Your Network 30
 - Role-Based Administration 32
 - ZfS Management Site 32
 - General ZfS Roles 33
 - ZfS Role-Based Modules and Roles 34
 - Configuring Role-Based Administration 45
 - Configuring Management and Monitoring Services 47
 - Stopping and Starting Management and Monitoring Services 47
 - Setting Up Discovery and Starting Back-End Processes 47
 - Setting Up the Alarm Management System 49
 - Setting Up Monitoring 49
 - Setting Up the Traffic Analysis Agent 49

- 2 Using ConsoleOne with Management and Monitoring Services 51**
 - Navigating the ZfS Namespace 51
 - Selecting ZfS Options 52
 - Views 53
 - Properties 53
 - Actions 54
 - Working with Views 54
 - Changing the Appearance of a View 54
 - Modifying Columns 55
 - Filtering Views 56
 - Sorting Views 57
 - Printing a View 57
 - Exporting a View 58
 - Saving Views 58
 - Deleting and Renaming Custom Views 59

- 3 Understanding Network Discovery and Atlas Management 61**
 - Understanding Network Discovery 62
 - Discovery Components 62
 - Discovery Process 68
 - What Is Discovered 75
 - File-Based Discovery 83
 - Effects of Discovery on Maps 86
 - Setting Up Discovery 88
 - Starting Discovery 90
 - Checking the Status of Initial Discovery 90
 - Checking the Results of Discovery 91

Changing the Default Configuration	92
Configuring the Java Processes	97
Unloading the Management Server	98
Managing the Atlas	99
Using the Atlas	99
Using Unified Views	101
4 Understanding Alarm Management	105
Understanding the Alarm Management System	105
Alarm Management System Components	106
Managing the Alarm Management System	109
Recognizing Alarm Indicators	109
Viewing Alarms	109
Enabling and Disabling Alarms	113
Resolving Alarms	114
Deleting Alarms	116
Performing Actions on Alarms	118
Maintaining the Alarm Management System	123
Troubleshooting the Alarm Management System	123
5 Understanding Server Management	125
Understanding Server Management	126
SNMP-Based Server Management	127
SNMP Agent Functions	127
Planning for Server Management	128
Creating a Baseline of Typical Server Activity	128
Using the Baseline Document	129
Server Baseline Document Tips	129
Optimizing Server Management	130
Setting Default Trends and Thresholds	130
Controlling Alarm Generation	134
Defining Recipients for SNMP Alarms	136
Managing Servers	137
Displaying Server Configuration Information	137
Displaying Summary Data	138
Viewing Trend Data	139
Managing Trend Samplings	141
Configuring Server Parameters	142
Executing Server Commands	143
Object Hierarchy and View Details	144
Object Hierarchy	144
Object View Details	146
6 Using the MIB Tools	165
Understanding MIB Tools	165
About MIBs	165
Understanding the SNMP MIB Compiler	165
Understanding the SNMP MIB Browser	166
Managing Devices with MIB Tools	168
Trap Definitions	169
Configuring MIBs and Setting Up MIB Tools	174
Annotating Third-Party MIBs for Integration with ZfS	174
Compiling MIBs for SNMP-Manageable Nodes	175
Using the MIB Browser	176
Browsing the MIB Tree	176
Viewing the Values of an Object and Its Child Nodes	177

Configuring a Node by Setting Object Values	178
Modifying SNMP Preferences.	179
Modifying Instances of an SNMP Table.	179
Forming Tables of Scalar Objects.	181
Graphing SNMP Request Results	182
Using a Profile for Tables and Graphs	184
Maintaining MIBs	184
7 Monitoring Services	185
Understanding Monitoring Services	185
Role-Based Services for Using the Monitoring Services	187
Monitoring Services on Target Nodes	187
Defining the Targets for Monitoring Services	187
Displaying Test Results Data	189
Changing the Test Options for a Node	190
Adding Services for Monitoring	190
8 Understanding Traffic Analysis	191
Understanding Traffic Analysis	191
Traffic Analysis Components	191
Communication Between Traffic Analysis Components	192
Traffic Analysis Features	193
Traffic Analysis Fundamentals	194
Planning for Segment Monitoring.	204
Creating a Baseline of Typical Segment Activity	204
Using the Baseline Document	204
Segment Baseline Document Tips	205
Preparing to Analyze Network Traffic.	206
Selecting the Preferred RMON Agent.	206
Setting Up SNMP Parameters	207
Analyzing Network Traffic.	208
Analyzing Traffic on Segments	209
Analyzing Traffic on Nodes Connected to a Segment	216
Capturing Packets	223
Displaying Captured Packets	226
Analyzing Traffic Generated by Protocols in Your Network.	232
Analyzing Traffic on Switches.	235
Optimizing Traffic Analysis	237
Choosing Options to Display Stations on a Segment.	237
Choosing Options to Display Trend Statistics.	239
Choosing Options to Display the Top Nodes Graph	241
Choosing Statistics to Display in the Unified Port Traffic View	242
Choosing Options to Display a Captured Packet	243
Configuring Alarm Options from the Set Alarm Dialog Box	244
Configuring the Monitor Nodes for Inactivity View	246
Understanding the Traffic Analysis Agents	247
Using the Traffic Analysis Agent for NetWare	249
Planning to Install the Traffic Analysis Agent for NetWare	249
Optimizing the Traffic Analysis Agent for NetWare Performance	250
Using the Console Utility of the Traffic Analysis Agent for NetWare	257
Using the Traffic Analysis Agent for Windows NT/2000	263
Changes Made During Installation	263
Planning to Install the Traffic Analysis Agent for Windows NT/2000	264
Optimizing the Traffic Analysis Agent for Windows NT/2000	267
Using LANZCON	270

9	Customizing Agent Configuration	273
	Agent Files	273
	Management Agent for NetWare Files.	273
	Management Agent for Windows NT Server Files.	274
	Customizing the Management Agent for NetWare	275
	SERVINST.NLM Load Parameters	276
	HOSTMIB.NLM Load Parameters	276
	NTREND.NLM Load Parameters	277
	Customizing the Management Agent for Windows NT Server	278
	Configuring the Management Agent for Windows NT Server	278
	Third-Party Agent Configuration.	279
	Ensuring that Traps Are Received.	279
	Integrating Vendor-Specific SNMP Traps	279
10	Protocol Decodes Suites Supported by ZfS	281
	NetWare Protocol Suite	281
	Network File System Protocol Suite.	283
	Systems Network Architecture Protocol Suite.	283
	AppleTalk Protocol Suite	283
	TCP/IP Protocol Suite	285
11	ZENWorks Management and Monitoring Services Database	289
	Understanding the ZfS Database	289
	Running the Database	289
	Database Caching	289
	Backing Up the Database	290
	Backing Up the Topology/Alarm Database	290
	Changing Database Passwords.	290
12	Using Reports in Management and Monitoring Services	291
	Understanding Management and Monitoring Services Reports.	291
	About the Topology Reports	291
	About the Alarm Reports.	294
	About the Health Reports	295
	Managing Reporting.	296
	Managing the Topology Reports.	297
	Managing the Server Management Health Reports	297
13	Using SNMP Community Strings	303
	About SNMP Community Strings	303
	SNMP Security.	303
	Setting the SNMP Community Strings.	304
	Setting the SNMP Community String: NetWare Server	304
	Setting the SNMP Community String: ConsoleOne	306
	Setting Community Strings for an Individual Node.	306
	Setting the SNMP Community String: Windows NT	307
A	Documentation Updates	309
	November 8, 2004.	309
	June 27, 2003	309
	Understanding Alarm Management	310
	Using Reports in Management and Monitoring Services	310
	April 15, 2003	310
	Understanding Network Discovery and Atlas Management	310
	Understanding Server Management	311
	December 19, 2002	311

Understanding Network Discovery and Atlas Management.	311
September 27, 2002	311
Understanding Network Discovery and Atlas Management.	311
Understanding Alarm Management.	312
May 17, 2002	312
Using SNMP Community Strings	312
Understanding Server Management	312
Using the MIB Tools	313
Understanding Traffic Analysis	313
Protocol Decodes Suites Supported by ZFS	313

Part II Policy and Distribution Services

14 Configuring Policy and Distribution Services	317
Planning Your Distribution System	318
Overview of Policy and Distribution Services	318
Selecting Your Distributions.	320
Understanding Your Network Topology.	324
Are Additional Distributors Needed?	325
Other Subscribers To Be Installed?.	328
Determining the Distribution Flow.	329
Understanding Distribution Security.	331
Determining the Channels for the Distributions	333
Determining Subscribers' Subscriptions	334
Determining the Distribution Schedules.	335
Configuring Your Distribution System	337
Installing Additional Distributors, Databases, and Subscribers	337
Setting Up Distributors in a Mixed Network Operating System Environment	340
Setting Up Additional Distribution Security	340
Starting the Distributor Agents	341
Setting Up the Additional Databases	343
Configuring the Distribution Flow	344
Creating the Distributions and Related Channels.	345
Subscribing to the Distributions	347
Sending the Distributions	348
Managing Your Distribution System	349
Configuration Planning Worksheet	349
15 Novell iManager	359
Accessing the ZfS Management Role in iManager.	359
Managing Tiered Electronic Distribution	361
Creating TED Objects in iManager	361
Editing TED Object Properties in iManager.	362
Deleting TED Objects in iManager	362
Monitoring the Distribution Process.	363
Monitoring Specific Agents	364
Managing the TED Agents from the Remote Web Console	364
Managing the Policy/Package Agent from the Remote Web Console	366
Opening Multiple Remote Web Console Windows	368
Comparing the ZfS Management Role in iManager with ConsoleOne Capabilities	368
16 Tiered Electronic Distribution	371
Understanding Tiered Electronic Distribution.	371
Distribution Management through Tiered Electronic Distribution	372
The Basic Distribution Process	372
TED's eDirectory Objects	373

Relationships of the TED Objects	373
Physical Network Connections.	374
Distribution Flow Details	374
The ZfS Agents Used by TED	374
The Tiered Distribution Model	376
TED's Key Components	377
Common Distribution Tasks	377
Distributors	379
Understanding Distributors.	379
Understanding Distribution Routing	381
Creating Distributors	389
Configuring Distributors	389
Refreshing the Distributor	392
Deleting a Distributor Object and How Its Distributions Are Affected	392
Distributions	393
Understanding Distributions	393
Distributions Issues	396
Determining the Distributions	397
Creating a Distribution	402
Prioritizing Distributions	410
Deleting a Distribution	410
Handling Orphaned Distributions	411
Manually Importing and Exporting Distributions	412
Using the TED Distribution Wizard.	414
Channels	415
Understanding Channels	416
Creating and Configuring Channels	417
Forcing a Channel To Be Sent.	418
Subscribers	418
Understanding Subscribers	418
Creating Subscribers	420
Configuring Subscribers	420
Updating Subscriber Configurations	423
Associating Subscribers with Channels	424
Deleting Subscriber Objects That Are Part of a Distributor's Routing Hierarchy	424
Subscriber Groups	424
Understanding Subscriber Groups.	424
Creating and Configuring Subscriber Groups	425
External Subscribers	426
Understanding External Subscribers.	426
Using External Subscribers for Out-of-Tree Distributions	430
Creating and Configuring External Subscribers	432
Configuring Multiple TED Objects	433
Issues with Modifying Multiple TED Object Properties.	434
Modifying Multiple TED Object Properties	434
Property Tabs Available for Multiple-Object Modifications.	435
Sending Distributions	439
Understanding the Distribution Processes.	440
Forcing a Single Distribution To Be Sent	441
Sending Distributions Through Parent Subscribers	441
Sending Distributions Between Trees	442
TED Issues	442
Understanding Dependencies in TED	443
System Resources and Server Behavior	443
Controlling I/O Rates and Concurrent Distributions	444

Minimizing Messaging Traffic	445
Changing DNS Names or IP Addresses for TED Servers	446
When a TED Process Fails	446
Working Directories	447
NetWare Distributor Directories	448
NetWare Subscriber Directories	449
Windows NT Distributor Directories	450
Windows NT Subscriber Directories	450
UNIX Distributor Directories	450
UNIX Subscriber Directories	451
Editing the TEDNODE.PROPERTIES File	451
17 Server Policies	455
Understanding Server Policies	455
Configuration and Behavioral Management through Server Policies	455
Server Policies and Packages	456
Plural and Cumulative Policies	456
Configuration and Behavioral Policies	457
Server Policies Architecture	457
Enforcing Policies	460
Server Policy Descriptions	461
Creating a Policy Package	465
Creating a Policies Container	466
Creating a Policy Package Object	466
Configuring Server Policies	467
Compiling ZENTRAP.MIB	467
Configuring the Container Package Policy	467
Configuring Server Package Policies	469
Configuring Service Location Package Policies	469
Configuring Distributed Server Package Policies	475
Creating Custom Log Files Using Policies	486
Enabling Policies	487
Distributing Policies	487
Associating Policies	487
Associating a Policy Package to the Distributor Object	488
Associating the Distributor Object to a Policy Package	488
Scheduling Policies	488
Scheduling a Policy	488
Editing the Default Schedule	489
Viewing Effective Policies	489
Viewing Effective Policies for ZfS 3.0.2 Servers	489
Viewing Effective Policies for ZfS 2 Servers	490
Changing Policy Enforcement	490
Modifying a Policy That Is Being Enforced	490
Stopping a Specific Policy From Being Enforced	490
Removing Policy Enforcement for a Specific Subscriber	491
Stopping Enforcement of a Policy Package Distribution	491
18 Server Software Packages	493
Software Management through Server Software Packages	493
Understanding Server Software Packages	493
Understanding Server Software Packages and Components	494
Understanding Software Package and Component Configurations	494
Determining the Installation Order of Software Packages	495
Compiling Software Packages	496
Accessing Software Packages	496

Distributing Software Packages	497
Distributing Software Packages to a Cluster	498
Failure of Software Package Installations	498
Rolling Back Software Package Installations	498
Planning Server Software Packages	499
Which Files or Applications Do I Want to Distribute?	499
What Are the Software Package Components?	500
What Are the Minimum Requirements?	500
What Are My Software Package Management Options?	500
Setting Up Server Software Packages	505
Setting Up Multiple-Workstation Management for Server Software Packages	506
Creating a Server Software Package	509
Configuring the Server Software Package	510
Creating the Software Package Components	510
Configuring the Software Package Components	511
Compiling a Software Package	521
Distributing the Software Package	521
Converting Older Server Software Packages to ZFS 3.0.2	521
19 Desktop Application Distribution	525
Understanding Desktop Application Distributions	525
The Purpose of Desktop Application Distributions	525
Distributed Application Issues	527
Miscellaneous Issues	537
Requirements	538
Creating a Desktop Application Distribution	539
Understanding the Desktop Application Distribution Wizard	539
Creating the Distribution	540
Rebuilding Desktop Application Distributions	545
All Attributes Are Updated	545
Triggering a Rebuild	545
Cleaning Up Desktop Application Distribution Files	546
Sending Desktop Application Distributions Tree-To-Tree	547
20 Security in Policy and Distribution Services	549
Distribution Security Using Signed Certificates and Digests	549
Understanding Digests	550
Understanding Certificate Usage in Policy and Distribution Services	550
Important Points about Certificates	551
ConsoleOne User Rights and Certificate Copying	552
Certificate File Locations	552
Resolving Certificates	552
Handling Invalid Certificates	553
Certificate and Private Key Directories	557
Creating Security Certificates for Non-Encrypted Distributions	557
Manually Copying Certificates for Non-Encrypted Distributions	558
Distribution Security Using Encryption	558
Creating and Copying Encryption Certificates	558
Sending an Encrypted Distribution	560
Extracting an Encrypted Distribution	561
Security for Inter-Server Communication Across Non-Secured Connections	562
Terms Used in This Section	562
Security Certificates	563
Using SSL for the ZenCSServlet	563
Format of the Password File	564
TCP/IP Addresses and DNS Names	564

21 Scheduling	565
Understanding Scheduling in Policy and Distribution Services	565
Scheduling Issues	565
Scheduling Differences Between Policies and TED	566
Scheduling Conflicts with Other Software.	566
Randomly Dispatch Option Issues	566
Distributor Scheduling Issues	568
TED Object Scheduling Issues	568
Calculating Time Differences	568
Inactivating Distributions and Channels.	569
The Schedule Types	570
Daily	571
Event	571
Interval	571
Monthly	571
Never	572
Package Schedule	572
Relative	572
Run Immediately	572
Time.	573
Weekly	573
Yearly	573
Scheduling Server Policies	573
Scheduling the TED Objects	574
Precedence of the Tiered Electronic Distribution Policy	574
TED Object Schedules	574
Using Intervals and Repeating Actions in Schedule Types.	577
Using Intervals with Distributors.	578
Repeating Actions.	578
22 Variables	579
Understanding Variables	579
Distribution Variable Example.	579
Where Variables Can Be Used	580
Types of Variables	580
Predefined Variables	580
User-Defined Variables	581
Resolution of Variable Names	581
Nested Variables	582
Creating a Variable	582
Creating Default Variables for All Subscribers	582
Creating Variables for a Specific Subscriber	583
Creating Variables for a Software Package.	583
Using a Variable to Change a Subscriber's Console Prompt.	584
Using Variables to Control File Extraction	584
23 ZENworks Database	587
Understanding the ZENworks Database	587
The Database File	587
Database File Location	587
The Database Object	587
Running the Database	588
Database Caching	588
Database Information	588
Coexisting Databases.	589
Determining How Many Databases You Need	589

Database Logging and TED Reporting	590
Multiple Databases	590
Installing, Setting Up, and Connecting To the ZENworks Database	592
Installing the Database	592
Configuring the ZENworks Database Policy	594
Connecting to the Database	595
Creating a ZENworks Database Object	596
Purging the Database	597
24 Reporting	599
Storing Report Information	599
Reporting Scope for TED Objects	600
Reporting on the Successes and Failures of Distributions	600
Generating Reports	600
Report Descriptions	601
TED Reports	601
Server Policy Reports	603
Creating Customized Reports	605
Default Sybase Database User ID and Password	605
Server Policies Database Contents	606
TED Database Contents	612
B Server Console Commands	617
ZfS Console Commands	617
Java Console Commands	620
C Load/Unload Actions	623
Load NLM/Process	623
Load Java Class	623
Unload Process	624
Start Service	624
Stop Service	624
D Requirements for Server Software Packages	625
Operating System	625
Memory (RAM)	626
Disk Space	626
SET Commands	627
Registry	627
File	628
PRODUCTS.DAT	628
E Registry Entries for Server Software Package Components	629
Key	629
Binary	630
Expand String	630
(Default)	630
DWord	631
Multi-Value String	631
String	631
F Using Server Software Packages to Delete Directories on Servers	633
Setting Up Variables for Use With the Server Software Package	633
Creating the Server Software Package	634
Creating and Configuring the Server Software Package Component	634
Compiling the Server Software Package	636
Manually Testing that the Directories Have Been Deleted	636

In Summary	636
G Documentation Updates	637
November 8, 2004	637
Desktop Application Distribution	638
July 29, 2004	638
Tiered Electronic Distribution	638
Server Policies	638
Server Console Commands	638
March 10, 2004	639
Tiered Electronic Distribution	639
June 27, 2003	639
Policy and Distribution Services	639
Tiered Electronic Distribution	640
Server Policies	641
Desktop Application Distribution	641
Security in Policy and Distribution Services	641
April 15, 2003	641
Configuring Policy and Distribution Services	642
Tiered Electronic Distribution	642
Server Software Packages	642
Security in Policy and Distribution Services	643
Reporting	643
Appendixes: Server Console Commands	643
Appendixes: Requirements for Server Software Packages	643
December 19, 2002	643
Configuring Policy and Distribution Services	644
Tiered Electronic Distribution	644
Server Software Packages	645
Desktop Application Distribution	645
Variables	645
ZENworks Database	646
Appendixes: Using Server Software Packages to Delete Directories on Servers	646
September 27, 2002	646
Policy and Distribution Services	646
Configuring Policy and Distribution Services	646
Novell iManager	647
Security in Policy and Distribution Services	647
June 6, 2002	647
Planning a Policy and Distribution Services Configuration	647
Understanding Security in ZENworks for Servers	648
May 17, 2002	648
Planning the Configuration	648
Configuring Policy and Distribution Services	648
Managing Your Distribution System	649
Automating Server Software Installations and Updates	649
Understanding Security in ZENworks for Servers	649
Part III Server Inventory	
25 Understanding Server Inventory	653
Server Inventory Terminology	653
Overview of Server Inventory Components	654
Inventory Scanners	654
Inventory Components on Inventory Servers	654
Inventory Database	655

Management Console	655
Understanding the Inventory Scanning Cycle in the Standalone Scenario	656
Understanding Rolling Up Scan Data Across Servers	657
26 Setting Up Server Inventory	661
Understanding the Inventory Server Roles	661
Root Server	661
Root Server with Inventoried Servers	663
Intermediate Server	664
Intermediate Server with Database	665
Intermediate Server with Inventoried Servers	666
Intermediate Server with Database and Inventoried Servers	667
Leaf Server	668
Leaf Server with Database.	669
Standalone Server.	670
Deploying Server Inventory	670
Deploying Server Inventory in a LAN Environment	670
Deploying Inventory over a WAN Environment	671
Possible Inventory Server Configurations for a WAN	676
Understanding the Effects of Server Inventory Installation	683
Setting Up the Inventory Database	684
Setting Up the Inventory Database for Sybase	685
Setting Up the Inventory Database for Oracle8i and Oracle9i.	690
Setting Up the Inventory Database for MS SQL Server 2000	701
Configuring Inventory Servers for Server Inventory	705
Configuring the Inventory Service Object	706
Configuring the Server Inventory Policy	707
Configuring the Database Location Policy.	710
Configuring the Roll-Up Policy.	711
Starting and Stopping the Inventory Service	712
Starting the Inventory Service	712
Stopping the Inventory Service	713
Changing the Role of the Inventory Server	713
Changing the Role of the Root Server.	715
Changing the Role of the Root Server with Inventoried Servers.	716
Changing the Role of the Intermediate Server.	717
Changing the Role of the Intermediate Server with Database.	718
Changing the Role of the Intermediate Server with Database and Inventoried Servers.	719
Changing the Role of the Intermediate Server with Inventoried Servers	720
Changing the Role of the Leaf Server	721
Changing the Role of the Leaf Server with Database	722
Changing the Role of the Standalone Server	722
27 Understanding the Server Inventory Components	725
Understanding the Inventory Service Manager	725
List of Services.	725
Services on NetWare Inventory Servers.	728
Services on Windows NT/2000 Inventory Servers.	729
Understanding the Server Configuration Service	730
Understanding the Inventory Scanner.	730
How the Scanners Collect server Inventory Data	731
Scanning Process Flowchart.	733
Software Information Collected by the Scanners	733
DMI-Compliant Scanners	734
WMI-Compliant Scanners	735
SNMP-Compliant Scanners	736

Hardware Data Collected by the Scanners	736
Understanding the Sender-Receiver	751
Understanding the Sender	753
Understanding the Receiver	754
Understanding the Compressed Scan Data File	754
Sender-Receiver Directories	755
Understanding the Selector.	756
Understanding the Storer.	757
Understanding the Inventory Removal Service.	757
Using the Inventory Removal Service for Synchronization	758
Understanding the Upgrade Service	759
An Overview of the Inventory Components on the Inventory Server	760
Understanding the Inventory Database.	760
Understanding ZfS Inventory Attributes.	760
28 Understanding the ZENworks for Servers Inventory Database Schema	787
Overview	787
CIM Schema	788
CIM-to-Relational Mapping	790
Logical Schema	792
Inventory Database Schema in ZfS	798
Case Study of CIM Schema Implementation in ZfS.	799
Legends for Schema Diagrams	801
CIM Classes and Extension Classes in ZfS.	801
Schema Diagrams of CIM and the Extension Schema in ZfS	803
Sample Inventory Database Queries	808
29 Managing Inventory Information	811
Viewing the Inventory Servers Deployed for Inventory	811
Viewing the Inventory Information	812
Configuring the Inventory Database	812
Viewing the Inventory Summary of an Inventoried Server	813
Viewing Inventory Information of Inventoried Servers by Querying the Database	821
Running Inventory Reports	824
Customizing the Inventory Information	831
Customizing Software Scanning of Inventoried Servers	831
Scanning for Vendor-Specific Asset Information from DMI	833
Customizing the Software Scanning Information of Vendors and Products	835
Customizing the Hardware Scanning Information of Jaz and Zip Drive Vendors	837
Exporting the Inventory Data to CSV Format.	837
Invoking the Data Export Tool	838
Exporting the Inventory Data to a CSV File.	838
Forming the Query and Setting the Filter Conditions	839
Loading an Existing Configuration File	841
Running the Data Export Program from the Inventory Server	842
30 Monitoring Server Inventory Using Status Logs	843
Viewing the Scan Status of an Inventoried Server	843
Viewing the Roll-Up History of the Inventory Server	844
Viewing the Status of Inventory Components on an Inventory Server	844
Viewing the Status of the Last Scan on the Inventoried Server	845
Viewing the Roll-Up Log for the Inventory Servers	845
Exporting the Inventory Status Log Files	846
Overview of Status Logs and Scan Logs	846
Viewing the Status Log in XML Format.	847

H	Documentation Updates	849
	March 11, 2005	850
	Setting Up Workstation Inventory	850
	Understanding the Server Inventory Components.	850
	November 8, 2004.	850
	March 10, 2004	851
	Setting Up Server Inventory	851
	Understanding the Server Inventory Components.	852
	October 17, 2003	852
	Setting Up Server Inventory	852
	Managing Inventory Information	852
	June 27, 2003	853
	Setting Up Server Inventory	853
	Managing Inventory Information	853
	December 18, 2002	854
	Understanding the Server Inventory Components.	854
	Managing Inventory Information	854
	September 27, 2002.	855
	Understanding Server Inventory	855
	Setting Up Server Inventory	855
	Understanding the Server Inventory Components.	856
	Managing Inventory Information	856
	June 18, 2002	857
	Setting Up Server Inventory	857
	May 17, 2002	858
	Setting Up Server Inventory	858
	Understanding Server Inventory	858
	Monitoring Server Inventory Using Status Logs	858

Part IV Remote Management

31	Remote Management for NetWare Servers	861
	Introduction	861
	RConsoleJ Client	861
	RConsoleJ Agent	861
	RConsoleJ Proxy Agent	861
	Deploying	862
	Setting Up RConsoleJ	862
	Initiating RConsoleJ	863
	Setting Up Security for RConsoleJ.	866
32	Remote Management for Windows NT/2000 Servers	867
	Remote Management Terminology	867
	Understanding Remote Management for Windows NT/2000 Servers.	867
	Setting Up Security for Remote Management.	869
	Configuring the Remote Management Policies	869
	Creating the Policy Packages	870
	Creating and Configuring the TED Objects	870
	Configuring the Server Remote Management Policy	871
	Configuring the Distribution Object for Remote Management	871
	Configuring the Distributor and the Subscriber Objects	872
	Setting Up the Agent Password at the Managed Server.	872
	Managing Remote Windows NT/2000 Servers	872
	Initiating Remote Management Sessions	872
	Managing a Remote View Session	874
	Managing a Remote Control Session	876

Viewing the Audit Log for Remote Management Sessions	883
Improving the Remote Management Performance	883
Unloading and Reloading the Remote Management Agent.	884

I Documentation Updates 885

November 8, 2004	885
May 17, 2002	885
Remote Management for Windows NT/2000 Servers	885

About This Guide

This guide describes how to administer Novell® ZENworks® for Servers (ZfS) 3.0.2. The guide is intended for network administrators and is divided into the following sections:

- ◆ “Management and Monitoring Services” on page 23
- ◆ “Policy and Distribution Services” on page 315
- ◆ “Server Inventory” on page 651
- ◆ “Remote Management” on page 859

Additional Documentation

For documentation on installing and running ZfS 3.0.2, see the [ZENworks for Servers 3.0.2 Installation guide](http://www.novell.com/documentation/lg/zfs302/index.html) (<http://www.novell.com/documentation/lg/zfs302/index.html>).

For documentation on troubleshooting ZfS 3.0.2, see the [ZENworks for Servers 3.0.2 Troubleshooting guide](http://www.novell.com/documentation/lg/zfs302/index.html) (<http://www.novell.com/documentation/lg/zfs302/index.html>).

Documentation Updates

For a dated list of updates to this guide, see:

- ◆ **Management and Monitoring Services:** Chapter A, “Documentation Updates,” on page 309
- ◆ **Policy and Distribution Services:** Appendix G, “Documentation Updates,” on page 637
- ◆ **Server Inventory:** Chapter H, “Documentation Updates,” on page 849
- ◆ **Remote Management:** Chapter I, “Documentation Updates,” on page 885

For the most recent version of the ZfS 3.0.2 guides, see the [ZENworks for Servers 3.0.2 documentation Web site](http://www.novell.com/documentation/lg/zfs302/index.html) (<http://www.novell.com/documentation/lg/zfs302/index.html>).

Documentation Conventions

In this documentation, a greater-than symbol (>) is used to separate actions within a step and items within a cross-reference path.

A trademark symbol (®, ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party Administration trademark.

When a single pathname can be written with a backslash for some platforms or a forward slash for other platforms, the pathname is presented with a backslash. Users of platforms that require a forward slash, such as UNIX*, should use forward slashes as required by your software.

Management and Monitoring Services

Novell® ZENworks® for Servers (ZfS) Management and Monitoring Services provides industry-standards-based monitoring, management, and reporting for heterogeneous network environments, including support for multi-protocol LAN/WAN networks and servers.

In addition, ZfS Management and Monitoring Services help you to proactively manage your NetWare® and Windows* NT* servers by responding faster to network problems and increasing overall system availability.

Management and Monitoring Services has the following components:

- ◆ **ConsoleOne®**, which provides the interface where you can manage and administer your network.
- ◆ **Management Site Services**, including:
 - ◆ Alarm Management
 - ◆ Database Administration
 - ◆ MIB Tools Administration
 - ◆ Monitoring Services
 - ◆ Network Discovery
 - ◆ Reporting
 - ◆ Role-Based Services
 - ◆ Topology Mapping
- ◆ **Server Management** for monitoring all the servers in your network
- ◆ **Traffic Analysis** for monitoring all traffic on Ethernet, token ring, or Fiber Distributed Data Interface (FDDI) network segments

The Management and Monitoring Services documentation contains the following sections:

- ◆ [Chapter 1, “Configuring Management and Monitoring Services,” on page 25](#)
- ◆ [Chapter 2, “Using ConsoleOne with Management and Monitoring Services,” on page 51](#)
- ◆ [Chapter 3, “Understanding Network Discovery and Atlas Management,” on page 61](#)
- ◆ [Chapter 4, “Understanding Alarm Management,” on page 105](#)
- ◆ [Chapter 5, “Understanding Server Management,” on page 125](#)
- ◆ [Chapter 6, “Using the MIB Tools,” on page 165](#)
- ◆ [Chapter 7, “Monitoring Services,” on page 185](#)
- ◆ [Chapter 8, “Understanding Traffic Analysis,” on page 191](#)
- ◆ [Chapter 9, “Customizing Agent Configuration,” on page 273](#)

- ◆ Chapter 10, “Protocol Decodes Suites Supported by ZfS,” on page 281
- ◆ Chapter 11, “ZENWorks Management and Monitoring Services Database,” on page 289
- ◆ Chapter 12, “Using Reports in Management and Monitoring Services,” on page 291
- ◆ Chapter 13, “Using SNMP Community Strings,” on page 303
- ◆ Chapter A, “Documentation Updates,” on page 309

1

Configuring Management and Monitoring Services

To use ZENworks[®] for Servers (ZfS) Management and Monitoring Services effectively, you must correctly install and configure the components on your network. You should have already performed a basic installation of ZfS (see [“Installing and Setting Up Management and Monitoring Services”](#) in the *ZfS Installation* guide).

The following sections provide you with the concepts and instructions to help you configure ZfS so that you can use its features to manage your network:

- ◆ [“Understanding Management and Monitoring Services” on page 25](#)
- ◆ [“Planning the Configuration” on page 29](#)
- ◆ [“Role-Based Administration” on page 32](#)
- ◆ [“Configuring Management and Monitoring Services” on page 47](#)

Understanding Management and Monitoring Services

This guide provides information on understanding, planning, managing, and monitoring ZfS Management and Monitoring Services. This section provides information about the components of the ZfS Management and Monitoring Services.

Management and Monitoring Services contains the following components:

- ◆ [“Management Site Services” on page 25](#)
- ◆ [“Server Management” on page 28](#)
- ◆ [“Traffic Analysis” on page 28](#)
- ◆ [“ConsoleOne” on page 29](#)

Management Site Services

The Management Site Services include the following:

- ◆ [“Network Discovery” on page 26](#)
- ◆ [“Database Administration” on page 26](#)
- ◆ [“Alarm Management” on page 26](#)
- ◆ [“Role-Based Services” on page 26](#)
- ◆ [“Reporting” on page 27](#)
- ◆ [“Topology Mapping” on page 27](#)
- ◆ [“MIB Tools Administration” on page 28](#)

- ◆ [“Monitoring Services” on page 28](#)

Network Discovery

When network autodiscovery is started, the servers, routers, switches which are SNMP instrumented, and the services hosted on these devices and workstations, are automatically discovered. The discovered data is written to a .DAT file and displayed in the atlas map on ConsoleOne.

Maps reflect the scope of discovery set at the management server. By default, all devices that the management server is able to establish communication with, are discovered and stored at the management server. By defining the scope of NetExplorer™, you can limit the number of discovered objects.

For more detailed information on network discovery, see [Chapter 3, “Understanding Network Discovery and Atlas Management,” on page 61](#).

Database Administration

ZfS provides a centralized Common Information Model (CIM)-compliant Sybase* database on the management server. The database serves as a repository for server and network data that can be displayed or formatted in various ways to provide you with the information you need to manage your network. The ZfS data is stored in a topology database containing three logical databases:

- ◆ Topology
- ◆ Alarms
- ◆ Map information

Most database functions are automatic and require very little administration. For more detailed information on ZfS databases, see [Chapter 11, “ZENWorks Management and Monitoring Services Database,” on page 289](#).

Alarm Management

Alarms recognized by ZfS include Simple Network Management Protocol (SNMP) traps, connectivity testing, and threshold profiling. Alarm management processes traps and proprietary alarms and forwards the alarms to ConsoleOne that subscribe to the alarms.

You can perform specific actions on an alarm by specifying the action in the alarm disposition. Some actions, like executing a program, sending an e-mail notification, and creating an archive, audible beep at the Console, and ticker messages, are automatically performed. You can set an action to forward specific processed alarms to other ZfS management servers, as well as forward unprocessed SNMP traps directly to a target address of any third-party enterprise management application.

Role-Based Services

ZfS Management and Monitoring Services supports role-based administration and task management through Novell eDirectory™. ZfS uses role-based services (RBS) to organize ZfS Management and Monitoring Services tasks into roles and to assign scope information to a role.

RBS roles specify tasks that users are authorized to perform. Defining an RBS role includes creating an RBS role object and specifying the tasks that the role can perform.

For general information on creating RBS role objects or specifying tasks that RBS roles can perform, see [“Configuring Role-Based Administration”](#) on page 45.

For information on how ZfS implements role-based services, see [“Role-Based Administration”](#) on page 32.

Reporting

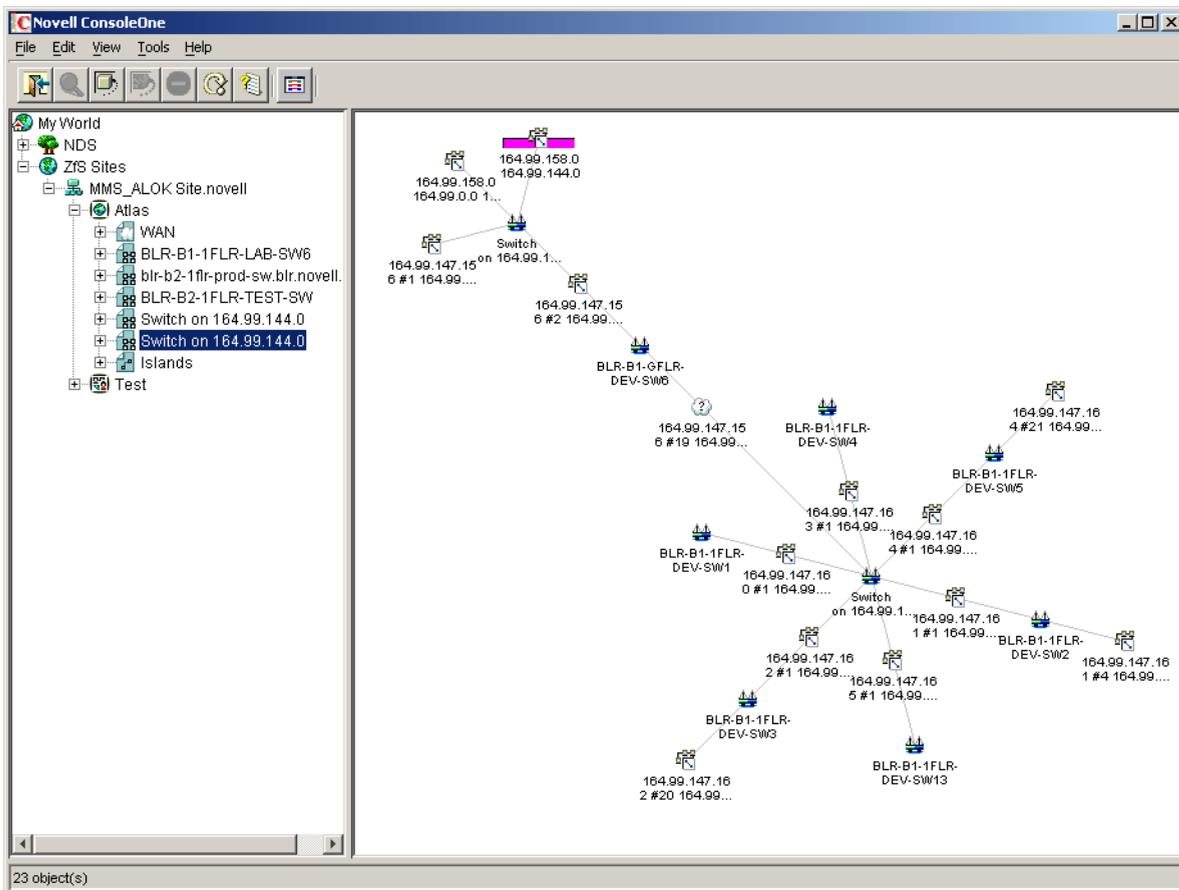
ZfS provides reporting services to generate statistical information. These reports can be displayed on ConsoleOne or exported to databases and Web formats. ZfS allows you to generate the following types of reports:

- ◆ Health reports
- ◆ Topology reports
- ◆ Alarm reports

For more detailed information on ZfS Management and Monitoring Services reports, see [“Using Reports in Management and Monitoring Services”](#) on page 291.

Topology Mapping

Topology mapping enables you to display maps in the ZfS hierarchical atlas as shown in the figure below. Maps reflect the scope of discovery set at the management server.



For more detailed information on topology mapping, see [“Managing the Atlas”](#) on page 99.

MIB Tools Administration

ZfS includes the MIB compiler and MIB browser, to manage SNMP devices.

The MIB tools enable you to:

- ◆ Set alarm templates for receiving SNMP traps
- ◆ Display and set values on SNMP devices
- ◆ Update trap definitions in the alarm template database
- ◆ Annotate third-party MIBs

For more detailed information on the MIB tools, see [Chapter 6, “Using the MIB Tools,” on page 165](#).

Monitoring Services

Monitoring, or SNMP, services include testing the connectivity and availability of a service on a network device. ConsoleOne is notified whenever the status of the service changes. The services that can be monitored include DHCP, DNS, Echo, FTP, HTTP, HTTPS, IP, IPX™, NFS, NNTP, SMTP, SNMP, Time Service, TFTP, and WUser.

For more detailed information on monitoring services, see [Chapter 7, “Monitoring Services,” on page 185](#).

Server Management

The server management component enables you to monitor all the servers in your network. This component must be installed on each of the servers you want to monitor using ConsoleOne. During the ZfS installation you can select the servers to install the server management component.

You can deploy some or all of the server monitoring software components to meet your management needs best. For more detailed information on server management, see [“Understanding Server Management” on page 125](#).

Traffic Analysis

The traffic management component provides the traffic analysis services for a NetWare or Windows NT server, to monitor all traffic on an Ethernet, Fiber Distributed Data Interface (FDDI), or token ring network segments.

The traffic analysis services include:

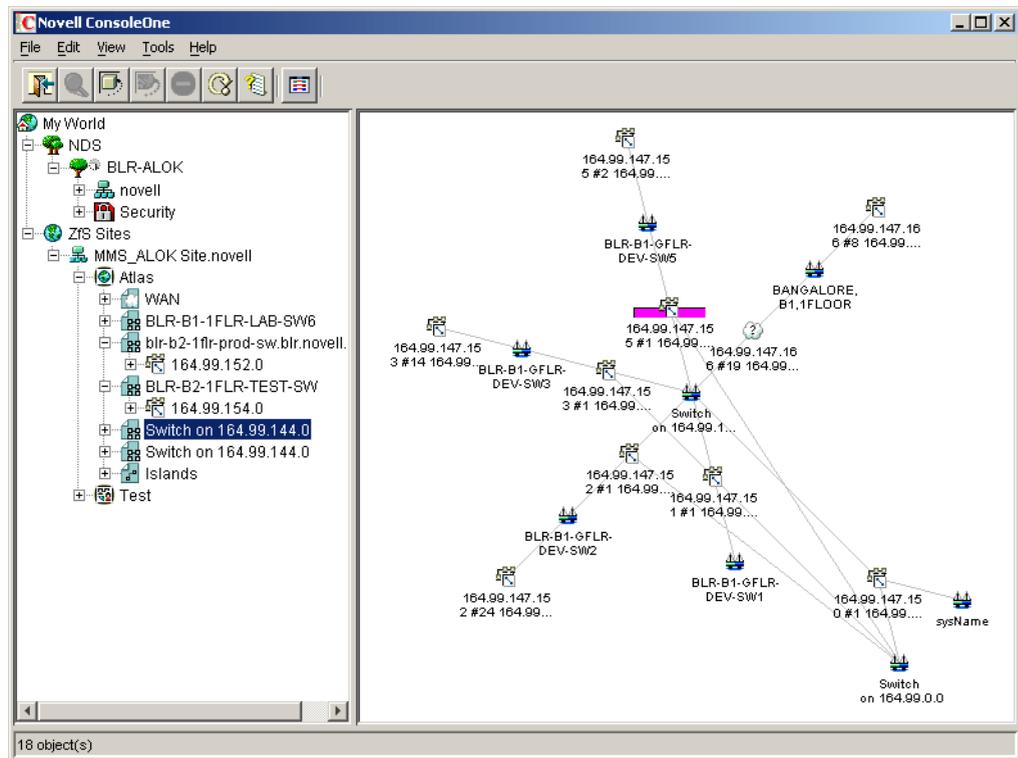
- ◆ Standard and enterprise-specific RFC 1757 MIB descriptions for remote network monitoring
- ◆ Extensions added to eDirectory, including Remote Monitor (RMON) agent configuration
- ◆ Network traffic trending and analysis tools
- ◆ Network health report templates
- ◆ Integration with topology maps
- ◆ Performance threshold configuration and profiling
- ◆ A view of conversations on network segment and utilization
- ◆ Packet capture tools and view

You can deploy some or all of the traffic analysis software components to meet your management needs best. For more detailed information on analyzing the network traffic, see [Chapter 8, “Understanding Traffic Analysis,”](#) on page 191.

ConsoleOne

The Novell ConsoleOne, provides the interface where you can manage and administer your network. ConsoleOne hosts programs (snap-ins) for integrating network administration and management snap-ins, enabling you to manage your network through a single interface.

ZfS provides a graphical user interface (GUI) snap-in to the Novell ConsoleOne under the ZENworks for Servers namespace, as shown in the following figure. It provides access to the unique functions provided by ZfS.



For more information on Novell ConsoleOne, see the [ConsoleOne Web site \(http://www.novell.com/products/netconsole/consoleone\)](http://www.novell.com/products/netconsole/consoleone).

Planning the Configuration

This section discusses general planning options for configuring the Management Site Services and some of the ZENworks for Servers (ZfS) agents (alarms, servers, and traffic) on your network. This section also discusses how to plan and implement role-based administration.

Before installing the ZfS Management and Monitoring Services software, you must decide what information you need to manage your network effectively. This section contains the following topics to help you decide the kind of information you would need to manage your network.

This section also explains how to configure the Management and Monitoring Services.

- ◆ “Defining Management Information Needs” on page 30
- ◆ “Planning a Strategy to Manage Your Network” on page 30
- ◆ “Configuring Your Network” on page 31

This guide also contains specific information on planning server management and segment monitoring in the following sections:

- ◆ “Planning for Server Management” on page 128
- ◆ “Planning for Segment Monitoring” on page 204

Defining Management Information Needs

ZfS is flexible to suit the business needs of different network configurations. You need to understand what information is needed by the groups in your organization and suitably deploy the software to meet those needs.

Typically, the groups in your company may consist of front-line help desk people, back-end information system administrators, and management-level coordinators, who need specific information for planning, budgeting, troubleshooting, and other issues.

For instance, one group might have a set of critical servers that need to be monitored round the clock. You might want real-time monitoring of these servers and receive notification when serious faults occur on these servers. Another example could be a need to generate weekly reports on server trends for a group of defined servers.

Planning a Strategy to Manage Your Network

In order for ZfS to monitor and manage devices on your network, it must actively poll your network segments and devices on your network. ZfS performs polling of these network objects using standard protocols (SNMP, TCP/IP, and IPX).

The design of the ZfS components minimizes the impact on network performance by storing trending information on the servers hosting the Simple Network Management Protocol (SNMP) and Remote Monitor (RMON) agents. Polling is directly performed by the management server based on requests coming from connected ConsoleOne.

The ZfS system administrator should configure the polling frequency to provide an appropriate level of monitoring for the network environment. A good rule for setting appropriate levels of monitoring is to identify systems that are critical for the operation. You can then group systems and segments into three basic management categories:

- ◆ **Mission critical:** Segments and devices that need to be actively monitored. Monitoring should be set at a high polling frequency.
- ◆ **Important:** Segments and devices that require less monitoring. These might be systems that host certain services that require a balance between polling overhead and performance. You should set the polling frequency to every few minutes, hours, or days.
- ◆ **Less important:** Segments and devices that require no active monitoring. Polling can be done on-demand to monitor segments and devices, or set to poll infrequently.

Devices that are either not polled or polled infrequently can be configured to send alarms (traps) to the management server to notify errors occurring on the system.

Configuring Your Network

The ZfS Management and Monitoring Services components rely on standard network protocols to communicate with devices on your network. In order to discover and accurately monitor your network and its devices, you need to ensure that the communication channels are consistent and well-configured.

The following sections discuss important aspects of your network configuration:

- ◆ [“IP Addressing Strategy” on page 31](#)
- ◆ [“IPX Transport Software” on page 31](#)
- ◆ [“eDirectory and DNS Name Resolution” on page 31](#)
- ◆ [“SNMP Configuration” on page 31](#)

IP Addressing Strategy

If you want to discover devices communicating over IP, ensure that they are configured with a valid IP address to enable you to manage the devices. TCP/IP must be bound on the designated ConsoleOne workstations and IP must be bound on the management server. You can use Dynamic Host Configuration Protocol (DHCP) addressing on ConsoleOne workstation, but a static address must be assigned to the management server.

IPX Transport Software

All devices communicating over IPX that you want to discover and manage must be configured with an IPX/SPX - compatible transport network software stack. NetWare and Windows drivers are included with the operating system installation software. ZfS is compatible with the Novell IP Compatibility Mode Driver.

eDirectory and DNS Name Resolution

Verify that your NetWare and Windows NT servers and network device names are in place before you begin discovering your network. Name resolution can be in the form of local host files, an eDirectory name, or a bindery table. The server names or host names are displayed in maps and configuration views rather than in IP or IPX addresses.

SNMP Configuration

The SNMP agents and RMON agents for Novell NetWare and Windows NT servers and other SNMP-enabled network devices require a community string to be identified on the device. You need to configure each SNMP-enabled device with a community string and trap target destination that includes that ZfS management server.

The community strings are used to ensure secure communication between the manager and the agents. In order for the ZfS system to communicate with an agent, the community string on the manager and agent must be similar and use the same port. In order to prevent all users from accessing information it is required to change the community string.

If the GET and SET community strings are changed from PUBLIC, you need to change settings at ConsoleOne and on the management server (load NXPCON > SNMP > Add/Edit Community Name) to match the names on your network. For details on how to change the community string, after installing the Management Services, see [“Changing the SNMP Community String” on page 93](#).

For information on configuring the NetWare and Windows NT server agents, see [Chapter 8, “Understanding Traffic Analysis,” on page 191](#).

Role-Based Administration

You can use ConsoleOne, a directory-enabled framework for running Novell network administration utilities. The ZfS snap-ins to ConsoleOne fully leverage eDirectory to enable role-based administration and higher levels of security. Through eDirectory, users will be able to log in once and have access to the management components as specified by their roles within their specific scope.

The ZfS snap-ins to ConsoleOne allows you to divide the task of network administration amongst administrators. With ConsoleOne, the functions and tasks of ZfS are organized into different, customized "views" based on each administrator's role in your organization.

The following sections discuss role-based administration:

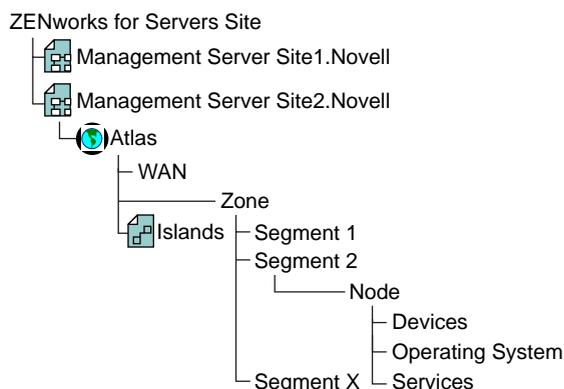
- ◆ [“ZfS Management Site” on page 32](#)
- ◆ [“General ZfS Roles” on page 33](#)
- ◆ [“ZfS Role-Based Modules and Roles” on page 34](#)
- ◆ [“Configuring Role-Based Administration” on page 45](#)

ZfS Management Site

The ZfS management site sets boundaries for accessing object data on the management server through the role-based services. You can create roles and tasks and further define the level of access to network objects and information from the network container space.

When you install ZfS Management and Monitoring Services, a management site, a system administrator role (RBS Admin), and all the site objects are created in eDirectory. A management site defines the scope of objects (networks, segments, routers, bridges, switches, servers, workstations, and so on) discovered on your network. You can create a single site or multiple sites, depending on the size of your network or network management requirements. A management site could include a single local network configuration or could encompass your entire network. The boundaries of a site are defined by the scope of network discovery. By default, network discovery is set to discover all connected networks and network nodes. The site object is created in the same context as the server object.

During installation, the default management site that is created is shown below. A single administration role is established with rights and permissions to all configuration and management tasks in the management system.



Some default roles that monitor network traffic, handle alarms, and manage server systems, are available and allow you to add users. You can also use them as examples for your new role creations.

In the ZfS role-based services (RBS), permissions that are required to access network objects, configurations, and information are associated with roles. eDirectory User objects can be assigned to appropriate roles. The levels of abstractions in a role are described below:

- ◆ Roles - Created to perform various network management functions in your organization. You can simplify granting of permissions and restrict access to management tools and data by creating appropriate roles.
- ◆ Tasks - Actions performed to utilize components of the management system based on the specific responsibilities.
- ◆ Component/module - A software tool that provides a network management function. ZfS includes components for managing servers, monitoring segment traffic, and providing common services such as database management, alarm handling, and report generation.

The users added to a role, however, retain the access rights, permissions, and policies granted through the eDirectory user account. For example, a user may be granted permission to access and configure a server through eDirectory, but may not be granted permission to manage the server through the RBS in ZfS. Therefore the management role that the user is assigned has limited access to the management services or components/modules in the ZfS management system.

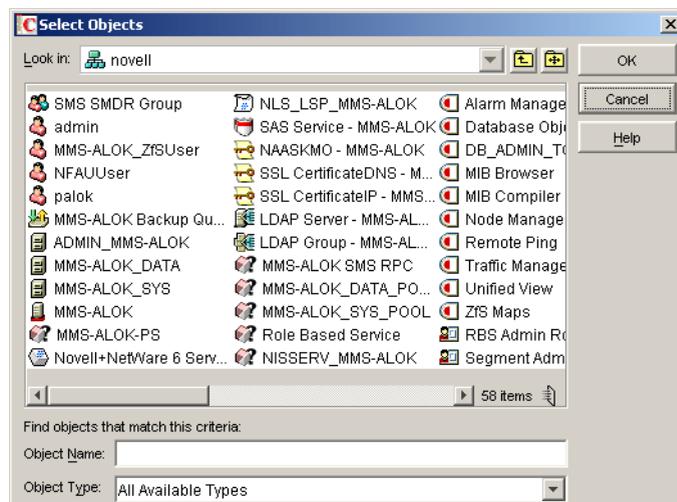
General ZfS Roles

ZfS components support role-based services (RBS) and task management through eDirectory. ZfS uses RBS to organize ZfS tasks into roles and to assign scope information to a role, user or a group.

RBS roles specify the tasks that users are authorized to perform. Defining an RBS role includes creating an RBS role object and specifying the tasks that the role can perform.

The tasks that RBS roles can perform are displayed as RBS Task objects in your eDirectory tree. These objects are organized into one or more RBS modules, which are containers that correspond to the different ZfS components. As shown in the figure below, ZfS provides predefined modules and RBS role objects.

IMPORTANT: You cannot create new modules or tasks. You have to select from the pre-defined modules and tasks that are available.



You can create any role using the modules and tasks. Each module can have one or more tasks. For example, RBS defines the task for Monitoring Services as Enable Remote Ping. If this task is assigned to your role, you can use the Monitoring Services facility. For a list of the predefined ZfS modules and ZfS roles along with the associated tasks, see [“ZfS Role-Based Modules and Roles” on page 34](#).

For more information on creating role objects using tasks and modules, see [“Configuring Role-Based Administration” on page 45](#).

ZfS Role-Based Modules and Roles

This section provides the following tables:

- ◆ ZfS Role-Based Modules and Associated Tasks
- ◆ ZfS RBS Predefined Roles and Associated Tasks

The following table lists each ZfS RBS module and the tasks that can be performed for the module.

ZfS RBS Module	Associated Tasks
Alarm Manager	<ul style="list-style-type: none"> ◆ Add Alarm Note ◆ Assign Alarm ◆ Define Alarm Disposition ◆ Delete Alarm ◆ View Active Alarms ◆ View Active Alarm History ◆ View Alarm Summary
Database Object Editor	Database Object Editor
DB_Admin_Tool	<ul style="list-style-type: none"> ◆ DB_BACKUP ◆ Database Password Change
MIB Browser	Enable MIB Browser
MIB Compiler	Enable MIB Compiler

Zfs RBS Module	Associated Tasks
Node Management	<ul style="list-style-type: none"> ◆ Clearing a Connection ◆ Create Health Profiles ◆ Create Health Reports ◆ Delete Health Profiles ◆ Delete Health Reports ◆ Downing a Server ◆ Loading an NLM ◆ Mounting and Dismounting a Volume ◆ Read Only All ◆ Read Only All Tabular View ◆ Read Only Health Profiles ◆ Read Only Health Reports ◆ Read Only Homepage ◆ Read Only HostFileSystemView ◆ Read Only InstalledSoftwareView ◆ Read Only NetWareLoadableModuleView ◆ Read Only NetWareUserView ◆ Read Only NetworkPerformanceView ◆ Read Only NTDiskListview ◆ Read Only NTMemoryUsageView ◆ Read Only NTNetworkView ◆ Read Only NTPartitionView ◆ Read Only NTApadpterView ◆ Read Only NTConnectionListView ◆ Read Only NWDiskListView ◆ Read Only NWMemoryUsageView ◆ Read Only NWNetworkMediaView ◆ Read Only NWProtocolView ◆ Read Only NWFileListView

ZfS RBS Module	Associated Tasks
	<ul style="list-style-type: none"> ◆ Read Only NWPartitionView ◆ Read Only NWQueueJobsListView ◆ Read Only NWQueueListView ◆ Read Only NWWVolumeListView ◆ Read Only NWWVolumeSegmentView ◆ Read Only NWWVolumeUsageView ◆ Read Only NWRunningSoftwareView ◆ Read Only Set Parameter ◆ Read Only Trend ◆ Read Write All ◆ Read Write All TabularView ◆ Read Write Health Profiles ◆ Read Write Health Reports ◆ Read Write Set Parameter ◆ Read Write Trend ◆ Remote Controlling ◆ Restarting a Server ◆ Unloading an NLM
Remote Ping	Enable Remote Ping
Traffic Management	<ul style="list-style-type: none"> ◆ Adding_Nodes_For_InactivityMonitoring ◆ Adding_Protocols_For_ProtocolDirectory ◆ Capture_Packets ◆ Deleting_Nodes_For_Inactivity ◆ Deleting_Protocols_For_ProtocolDirectory ◆ Freeing Agent Resources ◆ Setting_Segment_Alarms ◆ View_Conversations ◆ View_LANZ_Agents ◆ View_Protocol_Directory ◆ View_RMON_Summary

ZfS RBS Module	Associated Tasks
	<ul style="list-style-type: none"> ◆ View_Segment_Alarms ◆ View_Segment_Dashboard ◆ View_Segment_Monitor_Nodes_For_Inactivity ◆ View_Segment_Protocol_Distribution ◆ View_Segment_Stations ◆ View_Segment_Summary ◆ View_Segment_Trends ◆ View_Switch_Port_Traffic ◆ View_Switch_Summary
Unified View	<ul style="list-style-type: none"> ◆ Unified View for Devices ◆ Unified View for Segments
ZfS Maps	<ul style="list-style-type: none"> ◆ Import ◆ Layout ◆ Print ◆ Rebuild ◆ Rename ◆ Save

The following table lists each predefined ZfS RBS and the specific tasks that can be performed for each of the roles.

Management and Monitoring Services Predefined RBS Role	Management and Monitoring Services RBS Module	Assigned Default Tasks
RBS_Administrator	All Modules	All available tasks

Management and Monitoring Services Predefined RBS Role	Management and Monitoring Services RBS Module	Assigned Default Tasks
Segment Administrator	Alarm Manager	<ul style="list-style-type: none"> ◆ View Alarm Summary ◆ View Active Alarms ◆ View Alarm History ◆ Assign Alarms ◆ Add Alarm Note
	DM_Admin_Tool	No available tasks
	MIB Browser	No available tasks
	MIB Compiler	Enable MIB Compiler
	Node Management	<ul style="list-style-type: none"> ◆ Read Only Health Profiles ◆ Read Only Health Reports
	Remote Ping	Enable Remote Ping
	Traffic Management	<ul style="list-style-type: none"> ◆ Adding_Nodes_For_InactivityMonitoring ◆ Adding_Protocols_For_ProtocolDirectory ◆ Capture_Packets ◆ Setting_Segment_Alarms ◆ View_Conversations ◆ View_LANZ_Agents ◆ View_Protocol_Directory ◆ View_RMON_Summary ◆ View_Segment_Alarms ◆ View_Segment_Dashboard ◆ View_Segment_Monitor_Nodes_For_Inactivity ◆ View_Segment_Protocal_Distribution ◆ View_Segment_Stations ◆ View_Segment_Summary ◆ View_Segment_Trends ◆ View_Switch_Port_Traffic ◆ View_Switch_Summary
	ZfS Maps	<ul style="list-style-type: none"> ◆ Layout ◆ Print
	Unified Views	Unified Views for Segments

Management and Monitoring Services Predefined RBS Role	Management and Monitoring Services RBS Module	Assigned Default Tasks
Segment Manager	Alarm Manager	<ul style="list-style-type: none"> ◆ Assign Alarms ◆ Define Alarms Disposition ◆ Delete Alarms ◆ View Alarm Summary ◆ View Active Alarms ◆ View Alarm History ◆ Add Alarm Note
	DM_Admin_Tool	No available tasks
	Database Object Editor	Database Object Editor
	MIB Browser	Enable MIB Browser
	MIB Compiler	Enable MIB Compiler
	Node Management	<ul style="list-style-type: none"> ◆ Create Health Profiles ◆ Create Health Reports ◆ Delete Health Profiles ◆ Delete Health Reports ◆ Read Write Health Profiles ◆ Read Only Health Profiles ◆ Read Write Health Reports ◆ Read Only Health Reports
	Remote Ping	Enable Remote Ping

Management and Monitoring Services Predefined RBS Role	Management and Monitoring Services RBS Module	Assigned Default Tasks
Segment Manager <i>continued</i>	Traffic Management	<ul style="list-style-type: none"> ◆ Adding_Nodes_For_InactivityMonitoring ◆ Adding_Protocols_For_ProtocolDirectory ◆ Capture_Packets ◆ Deleting_Nodes_For_InactivityMonitoring ◆ Deleting_Protocols_For_ProtocolDirectory ◆ Freeing Agent Resources ◆ Setting_Segment_Alarms ◆ View_Conversations ◆ View_LANZ_Agents ◆ View_Protocol_Directory ◆ View_RMON_Summary ◆ View_Segment_Alarms ◆ View_Segment_Dashboard ◆ View_Segment_Monitor_Nodes_For_Inactivity ◆ View_Segment_Protocol_Distribution ◆ View_Segment_Stations ◆ View_Segment_Summary ◆ View_Segment_Trends ◆ View_Switch_Port_Traffic ◆ View_Switch_Summary
	ZfS Maps	<ul style="list-style-type: none"> ◆ Import ◆ Layout ◆ Print ◆ Rebuild ◆ Rename ◆ Save

Management and Monitoring Services Predefined RBS Role	Management and Monitoring Services RBS Module	Assigned Default Tasks
Segment Monitor	Alarm Manager	<ul style="list-style-type: none"> ◆ View Alarm Summary ◆ View Active Alarms ◆ View Alarm History
	DM_Admin_Tool	No available tasks
	MIB Compiler	No available tasks
	MIB Browser	No available tasks
	Node Management	<ul style="list-style-type: none"> ◆ Read Only Health Profiles ◆ Read Only Health Reports
	Remote Ping	Enable Remote Ping
	Traffic Management	<ul style="list-style-type: none"> ◆ Capture_Packets ◆ View_Conversations ◆ View_LANZ_Agents ◆ View_Protocol_Directory ◆ View_RMON_Summary ◆ View_Segment_Alarms ◆ View_Segment_Dashboard ◆ View_Segment_Monitor_Nodes_For_Inactivity ◆ View_Segment_Protocol_Distribution ◆ View_Segment_Stations ◆ View_Segment_Summary ◆ View_Segment_Trends ◆ View_Switch_Port_Traffic ◆ View_Switch_Summary
	ZfS Maps	<ul style="list-style-type: none"> ◆ Layout ◆ Print
	Unified Views	Unified View for Segments

Management and Monitoring Services Predefined RBS Role	Management and Monitoring Services RBS Module	Assigned Default Tasks
Server Administrator	Alarm Manager	<ul style="list-style-type: none"> ◆ Assign Alarm ◆ Define Alarm Disposition ◆ Delete Alarm ◆ View Alarm Summary ◆ View Active Alarms ◆ View Alarm History ◆ Add Alarm Note
	DM_Admin_Tool	No available tasks
	MIB Browser	Enable MIB Browser
	MIB Compiler	No available tasks
	Node Management	<ul style="list-style-type: none"> ◆ Clearing a Connection ◆ Loading an NLM ◆ Mounting and Dismounting a Server Volume ◆ Downing a Server ◆ Read Only Health Profiles ◆ Read Only Health Reports ◆ Read Write All ◆ Restarting a Server ◆ Unloading an NLM
	Remote Ping	Enable Remote Ping
	Traffic Management	No available tasks
	ZfS Maps	<ul style="list-style-type: none"> ◆ Layout ◆ Print
	Unified Views	Unified Views for Devices

Management and Monitoring Services Predefined RBS Role	Management and Monitoring Services RBS Module	Assigned Default Tasks
Server Manager	Alarm Manager	<ul style="list-style-type: none"> ◆ Assign Alarm ◆ Define Alarm Disposition ◆ Delete Alarm ◆ View Alarm Summary ◆ View Active Alarms ◆ View Alarm History ◆ Add Alarm Note
	DM_Admin_Tool	No available tasks
	MIB Browser	No available tasks
	MIB Compiler	No available tasks
	Node Management	<ul style="list-style-type: none"> ◆ Clearing a Connection ◆ Create Health Profiles ◆ Create Health Reports ◆ Delete Health Profiles ◆ Delete Health Reports ◆ Downing a Server ◆ Loading an NLM ◆ Mounting and Dismounting a Server Volume ◆ Read Only Health Profiles ◆ Read Only Health Reports ◆ Read Write All ◆ Read Write Health Profiles ◆ Read Write Health Reports ◆ Restarting a Server ◆ Unloading an NLM
	Remote Ping	No available tasks
	Traffic Management	No available tasks
	ZfS Maps	<ul style="list-style-type: none"> ◆ Import ◆ Layout ◆ Print ◆ Rebuild ◆ Rename ◆ Save

Management and Monitoring Services Predefined RBS Role	Management and Monitoring Services RBS Module	Assigned Default Tasks
Server Manager <i>continued</i>	Database Object Editor	Database Object Editor
	Unified Views	Unified View for Devices
Server Monitor	Alarm Manager	<ul style="list-style-type: none"> ◆ View Alarm Summary ◆ View Active Alarms ◆ View Alarm History
	DM_Admin_Tool	No available tasks
	MIB Browser	No available tasks
	MIB Compiler	No available tasks
	Node Management	<ul style="list-style-type: none"> ◆ Read Only Health Profiles ◆ Read Only Health Reports ◆ Read Only Homepage ◆ Read Only HostFileSystemView ◆ Read Only InstalledSoftwareView ◆ Read Only NetWareLoadableModulesView ◆ Read Only NetWareUserView ◆ Read Only NetworkPerformanceView ◆ Read Only NTDiskListView ◆ Read Only NTMemoryUsageView ◆ Read Only NTNetworkView ◆ Read Only NWConnectionListView ◆ Read Only NWOpenListView ◆ Read Only NWDiskListView ◆ Read Only NWMemoryUsageView ◆ Read Only NWNetworkMediaView ◆ Read Only NWFileListView ◆ Read Only NWWolumeListView ◆ Read Only NWWVolumeUsageView ◆ Read Only RunningSoftwareView ◆ Read Only Trend
	Remote Ping	Enable Remote Ping
	Traffic Management	No available tasks
ZfS Maps	<ul style="list-style-type: none"> ◆ Layout ◆ Print 	

Management and Monitoring Services Predefined RBS Role	Management and Monitoring Services RBS Module	Assigned Default Tasks
Site Database Administrator	Alarm Manager	No available tasks
	DM_Admin_Tool	<ul style="list-style-type: none"> ◆ DB_BACKUP ◆ Database Password Change
	MIB Browser	No available tasks
	MIB Compiler	No available tasks
	Node Management	No available tasks
	Remote Ping	No available tasks
	Traffic Management	No available tasks
	ZfS Maps	No available tasks

Configuring Role-Based Administration

Defining an RBS role includes creating an RBS role object and specifying the tasks that the role can perform.

The following sections discuss how to configure Role- Based Administration:

- ◆ “Defining RBS Role” on page 45
- ◆ “Creating an External Scope” on page 46
- ◆ “Assigning RBS Role Membership and Scope” on page 46

Defining RBS Role

RBS roles specify the tasks that users are authorized to perform in specific administration applications. Defining an RBS role includes the following sections:

- ◆ “Creating an RBS Role Object” on page 45
- ◆ “Specifying the Tasks that RBS Roles Can Perform” on page 46

Creating an RBS Role Object

To create an RBS role object:

- 1** Right-click the container that you want to create the RBS role object > click New > click Object.
- 2** Under Class, select RBS:Role > click OK.
- 3** Enter a name for the new RBS role object.

Ensure to follow proper eDirectory naming conventions. For eDirectory naming conventions see [Novell eDirectory Administration Guide \(http://novell.com/documentation\)](http://novell.com/documentation).

Example: Password Administrator Role.

- 4** Click OK.

Specifying the Tasks that RBS Roles Can Perform

To specify the tasks:

- 1 Right-click an RBS role > click Properties.
RBS task objects are located only in RBS module containers
- 2 In the Role Based Services tab, make the associations you want.
- 3 Select the Role Content page > Add the list of tasks that the role can perform.
- 4 Click OK.

Creating an External Scope

To create an external scope:

- 1 Right-click the container that you want to create the scope object > click New > click Object.
- 2 Under Class, select MW:Scope > click OK.
- 3 Enter a name for the new MW:Scope object.
Ensure to follow proper eDirectory naming conventions. For eDirectory naming conventions see [Novell eDirectory Administration Guide \(http://novell.com/documentation\)](http://novell.com/documentation).
Example: Password Administrator Role.
- 4 Click OK.

Configuring a Scope Object

To configure a scope object:

- 1 Right-click the scope object > click Properties.
- 2 Browse the site object to which the scope is associated.
- 3 In the Site scope browse to select the computers to the site scope.
- 4 In the SQL script specify the scope by selecting the object and the operator from the drop-down list.
- 5 Click OK.

IMPORTANT: By default the scope object will have all-site access.

The effective scope will be a union of Site scope and the objects specified in SQL script.

Assigning RBS Role Membership and Scope

To assign an RBS role and scope to a user:

- 1 Right-click the user object to which you want to assign the role and scope > click Properties.
- 2 Click on Role Based Services Tab > Assigned Roles.
- 3 Click Add to add the required role to the user.
- 4 Click Scope to add the scope for the user.
- 5 Click OK.

IMPORTANT: If a user is assigned two different roles with different scopes, the user has rights to all the tasks (union of tasks in role1 and tasks in role2) irrespective of the scopes.

You cannot assign role and scope to User groups and Organization Unit.

Configuring Management and Monitoring Services

ZfS is made up of several components, some of which require certain setup tasks before you can use them, and others that do not.

The following components do not require any specific setup tasks:

- ◆ ZfS databases
- ◆ Role-based services (RBS)
- ◆ Management Information Base (MIB) tools
- ◆ ConsoleOne
- ◆ Reporting
- ◆ SNMP services

The following sections describe the setup tasks that are required to get the following components up and running:

- ◆ [“Stopping and Starting Management and Monitoring Services” on page 47](#)
- ◆ [“Setting Up Discovery and Starting Back-End Processes” on page 47](#)
- ◆ [“Setting Up the Alarm Management System” on page 49](#)
- ◆ [“Setting Up Monitoring” on page 49](#)
- ◆ [“Setting Up the Traffic Analysis Agent” on page 49](#)

Stopping and Starting Management and Monitoring Services

If you need to install other software or perform other maintenance functions on your server, you can stop Management and Monitoring Services and down the server. After performing the maintenance, you must reboot the server and restart the services in order for the server to resume its Management and Monitoring Services.

To stop and restart Management and Monitoring Services and down the server, complete the following steps at the management server console prompt:

- 1** To stop and unload ZfS Management and Monitoring Services, enter `unmw`.
- 2** To stop all JAVA processes, enter `java -killall`.
- 3** To exit JAVA, enter `java -exit`.
- 4** To down the server and restart, enter `restart server`.

To down the server, enter `down server`. You need to start the server again.

Because the appropriate commands to start the back-end and discovery processes (SLOADER and NETEXPLOR) were inserted in the AUTOEXEC.NCF file when you installed Management and Monitoring Services, restarting the server will start these processes. If you modified the AUTOEXEC.NCF file and need to manually start these processes, see [“Manually Starting Discovery and Back-End Processes” on page 48](#).

Setting Up Discovery and Starting Back-End Processes

The discovery software on the management server automatically discovers the nodes on your network. Network nodes include servers, desktops, routers, switches, and any other network

devices. Discovery starts automatically when the ZfS software is loaded on the management server and runs continually, 24 hours a day. The amount of time to build a complete database depends on the size of your network. Very small networks might take one or two hours; very large networks (several thousand nodes) might require several days.

It is recommended that you run Network discovery on a standalone as the discovery process consumes a longer duration if you use the system.

After installation, your servers are in one of the following states:

- ◆ Discovery and back-end services are running.

If you selected Yes to start the autodiscovery process and back-end services during installation, discovery is running on your ZfS server and your network is continually being discovered. You do not need to do anything further with regards to configuring discovery unless you want to modify your discovery parameters after you check the results of the initial discovery. For instructions on checking the results of discovery and modifying your discovery parameters, see [Chapter 3, “Understanding Network Discovery and Atlas Management,” on page 61](#).

IMPORTANT: After modifying any discovery parameters, you must restart the server as described in [“Stopping and Starting Management and Monitoring Services” on page 47](#).

- ◆ Discovery and back-end services are not running.

If you selected No, and did not start the autodiscovery process and back-end services during installation, you must start discovery after you modify the default discovery parameters. For specific instructions on modifying discovery parameters, see [Chapter 3, “Understanding Network Discovery and Atlas Management,” on page 61](#).

Before discovering your network, you can modify the following discovery parameters:

- ◆ SNMP Community Strings. Ensure that discovery is configured with the community strings of your devices.
- ◆ Discovery Scope. By default, discovery will discover the entire network if correct community strings are provided. If the discovery scope needs to be limited for some reason, it can be modified.
- ◆ IPX Discovery. IPX discovery will take place as long as the ZfS server has a valid IPX address binding. If there is no IPX address bound to the ZfS server, but there are IPX networks that need to be discovered, install the NetWare server in CMD mode (load SCMD).

IMPORTANT: After modifying any discovery parameters, you must restart the services as described in [“Stopping and Starting Management and Monitoring Services” on page 47](#). If you never started discovery or the back-end services, you can manually start the services as described in [“Manually Starting Discovery and Back-End Processes” on page 48](#).

Manually Starting Discovery and Back-End Processes

The commands to start autodiscovery and load the back-end services are inserted into the AUTOEXEC.NCF file by the installation program. Restarting the server will automatically start these processes. However, if you remove these commands you will need to manually start autodiscovery and load the back-end services (management site services).

During installation, a search path is added to the AUTOEXEC.NCF file to the management server program file path — ZENWorks\MMS\MWSERVER\BIN

Type the following in order at the management server console prompt, to manually start discovery and the back-end processes:

1. **mgmtddb** — This starts the Sybase database.
2. **mserver** — This starts the Naming service (MMSNAMING.NCF) and the Trap Receiving service (SNMPLOG.NLM).
3. **netxplor.ncf** — This starts the autodiscovery process.
4. **sloader.ncf** — This starts the basic services like Alarm Manager, Atlas Manager, Topology Manager, etc. The services to be started are listed in the SLOADER.PROPERTIES located in the ZENWorks\MMS\MWSERVER\PROPERTIES directory.

The server will accept requests from ConsoleOne only after the SLOADER.NCF is completely loaded.

Setting Up the Alarm Management System

The ZfS Alarm Management System (AMS) can receive SNMP traps from any SNMP-enabled device or computer hosting a proxy SNMP agent. If your network device is using Management Agent for NetWare, Management Agent for Windows NT, NetWare LANalyzer[®] Agent[™], or LANalyzer Agent for Windows NT software, the device is discovered automatically for you. No setup is needed after installing the software.

Third-party SNMP agents require some setup before traps can be received. For information on setting up third-party SNMP agents, see [“SNMP Configuration” on page 31](#).

Setting Up Monitoring

Because the Management Agent for NetWare and the ManageWise[®] Agent for Windows NT are based on SNMP, all actions that are directed from network management console to a server involve SNMP SET and GET requests from the manager to the agent. Any ConsoleOne requesting data from a managed server does so by issuing an SNMP GET request. An SNMP SET command is required to set server alarm thresholds or configuration parameters. Conducting these management operations from ConsoleOne such as ConsoleOne, raises the issue of ensuring security. In particular, unauthorized users setting configuration parameters on a server could cause severe performance problems or even sabotage network operations.

For these reasons, you should secure communication between the management system and your SNMP agents. For further information on SNMP security, [“SNMP Configuration” on page 31](#).

Setting Up the Traffic Analysis Agent

The Traffic Analysis Agent for NetWare is a distributed network analyzer that complements ZfS. While other ZfS agents collect data about specific network nodes, such as servers, the Traffic Analysis Agent for NetWare observes the interaction among these nodes on a specific LAN segment. The agent is installed on a NetWare 4.x or NetWare 5.x server. To set up Traffic Analysis Agent for NetWare, see [“Starting the Traffic Analysis Agent for NetWare” on page 50](#).

The Traffic Analysis Agent for Windows NT/2000 uses SNMP to communicate with the management server. After installation, in order for the Traffic Analysis Agent for Windows NT/2000 to operate, you must start the SNMP services. To start SNMP services, complete [“Starting the SNMP Service for the Traffic Analysis Agent for Windows NT/2000” on page 50](#).

After the agents are set up, you must restart the Windows NT/2000 server on which the agent resides.

Starting the Traffic Analysis Agent for NetWare

The installation program for the Traffic Analysis Agent for NetWare modifies the AUTOEXEC.NCF file so that the agent starts automatically. Therefore, you do not need any further configuration. If, however, you are upgrading from a previous version of the Traffic Analysis Agent (referred to as the LANalyzer agent), and did not uninstall the previous version, you must ensure that each server on which you upgraded the agent will run the new Traffic Analysis Agent.

To ensure that the upgraded NetWare servers run the new Traffic Analysis Agent:

- 1** On each NetWare server where you upgraded the ZfS Traffic Analysis Agent, open the AUTOEXEC.NCF file located in SYS:\SYSTEM.
- 2** Comment out the following lines by placing a # character at the beginning of the line as follows:

```
#Search add lanzdir
```

```
#LANZ.NCF
```

The first statement defines the search path where *lanzdir* is the directory in which the older agent is installed. The second statement loads the older agent.

- 3** Save the file and restart the server.

The new agent will load and run automatically. The LANZ.NCF file in the *agentinstallfolder*\LANZ will start the Traffic Analysis agent. The ULANZ.NCF in the same folder will stop the Traffic Analysis agent.

Starting the SNMP Service for the Traffic Analysis Agent for Windows NT/2000

If you have configured Windows NT/2000 to start the SNMP service automatically, the agent installed on Windows NT/2000 starts with the SNMP service when you start Windows NT/2000.

If you have not configured Windows NT/2000 to start the SNMP service automatically, do either of the following:

- ◆ At the command prompt, enter **net start snmp**.
- ◆ From the Control Panel, click Services > SNMP > Start.

When the SNMP service is started, the traffic analysis agent for Windows NT/2000 will also start.

2

Using ConsoleOne with Management and Monitoring Services

The ZENworks® for Servers (ZfS) console is a snap-in to the ConsoleOne® management tool. ZfS expands ConsoleOne management capabilities by adding menu options, property pages for existing Novell® eDirectory™ objects, and ways to browse and organize network resources. This section introduces ConsoleOne features that are unique to ZfS, including:

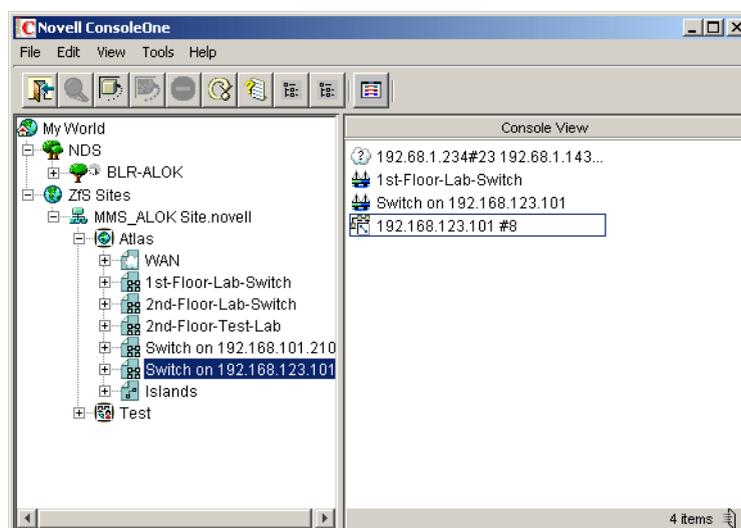
- ♦ “Navigating the ZfS Namespace” on page 51
- ♦ “Selecting ZfS Options” on page 52
- ♦ “Working with Views” on page 54

For more information on basic ConsoleOne capabilities, see the [Novell ConsoleOne Administration Guide \(http://novell.com/documentation\)](http://novell.com/documentation).

Navigating the ZfS Namespace

In ConsoleOne, your network and its resources are regarded as a set of objects and are arranged in various containers. Each top-level object is referred to as a namespace. To view your network and its resources on ConsoleOne, you must log in to the eDirectory tree which contains site server object.

The ZfS ConsoleOne snaps in to ConsoleOne under the ZfS Sites namespace, as shown in the following figure:

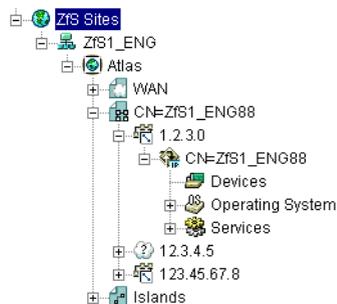


In general, you can perform administration tasks by browsing to an object in the left frame, right-clicking it, and clicking an option. Objects within the ZfS namespace are arranged in the following hierarchy:

1.  **ZENworks for Servers sites object:** This is the ZfS namespace container. It is the top of the ZfS namespace hierarchy. Expand this object to display a list of ZfS management sites.
2.  **ZfS Site:** This object represents a ZfS management server. It represents an eDirectory object that defines a collection of discovered objects that collectively make up a group of services. Expand this object to display the atlas for the site.
3.  **Atlas:** This is the container object for all discovered topology objects. The atlas can contain the following types of pages:
 - ♦  **WAN page:** Summarizes the entire network.
 - ♦  **Area page:** Displays segments on the network. There may be more than one Area page, depending on how your network is organized.
 - ♦  **Islands page:** Displays segments with undetermined connectivity.
4.  **Segments:** Within each atlas page is a listing of the segment objects that are included in that section of the atlas.
5.  **Nodes:** Within each segment object is a listing of server and node objects that reside on the segment. The icon displayed varies by the node type.
6. **Node Details:** Expand a node object to display a list of system internal components. Server data is grouped into the following three categories:
 - ♦  **Devices**
 - ♦  **Operating System**
 - ♦  **Services**

You can drill down into the server configuration further by clicking the plus signs next to the Devices, Operating System, and Services objects to display details about the internal components of the server. The internal components include the processors, installed software, volumes, kernel, and adapters associated with the server. For more details about the node objects, see [“Object Hierarchy” on page 144](#).

The following figure illustrates the ZfS namespace hierarchy:



Selecting ZfS Options

To display the ZfS options, you want to monitor or manage in the left frame, right-click the object. The options available are displayed. ZfS provides three main options:

- ◆ “Views” on page 53
- ◆ “Properties” on page 53
- ◆ “Actions” on page 54

Views

Views are different ways of displaying information. ZfS provides a variety of views designed to help you view the information of your network in different ways. The views ZfS provides are:

- ◆ **Atlas:** Provides a graphical representation of the discovered network topology, the physical location of nodes, node configuration, and alarm information.
- ◆ **Console:** Displays the objects contained in the selected container object. This view is useful while navigating the ZfS site.
- ◆ **Trend:** Provides a graphical representation of current and historical trend data by hour, day, week, month, or year. Monitoring trend data helps you with tasks such as determining which server is being used, who is using the server, troubleshooting problems, balancing load across multiple servers, and planning resources.
- ◆ **Active Alarms:** Provides a tabular display of alarm statistics for all the current alarms received from segments or devices, per management site. This view is refreshed whenever a new alarm occurs on the network.
- ◆ **Alarm History:** Provides a tabular display of all archived alarms, including the handled status of each alarm. This view is refreshed whenever a new alarm occurs on the network.
- ◆ **Alarm Summary:** Provides a graphical representation of the summary of alarms you have received. The view is divided into three panels of representation: pie chart panel, bar graph pane, and trend panel. Provides a tabular display of all archived alarms, including the handled status of each alarm.
- ◆ **Summary:** Provides a tabular information about the selected object’s configuration. For example, the summary view for a server object displays information about NLM™ files, memory usage, adapters, network interfaces, disks and disk controllers, volumes, queues, users, connections, open files, alarms, and installed software.

In addition to these main views, ZfS provides additional views for many of the objects in the hierarchy. For example, if you select a memory object, you can select a disk cache view that displays utilization for disk cache memory. For more information on the available views and the specific information displayed in an object view, see “Object View Details” on page 146.

Properties

The ZfS ConsoleOne provides several property pages that allow you to control ZfS-specific settings. To access the ZfS property pages, right-click an object and then click Properties.

- ◆ At the site level, ZfS provides property pages that allow you to edit global properties like Alarm Dispositions, ZfS Database settings, SNMP settings, MIB Pool entries, and health report profiles.
- ◆ At the server level, ZfS provides property pages that allow you to modify SNMP settings.

For general information on using ConsoleOne property pages, see the [Novell ConsoleOne Administration Guide \(http://novell.com/documentation\)](http://novell.com/documentation).

Actions

You can perform one or more actions on some objects. For example, if you right-click a server object, the Actions menu provides options for restarting or shutting down the server. However, if you right-click a volume object, the Actions menu provides options for mounting or dismounting the volume. For more information on performing actions on a managed object, see [“Executing Server Commands” on page 143](#).

Working with Views

ZfS ConsoleOne provides two types of views: tabular (list) views and graphical views. The Console, Active Alarms, and Alarm History views are all tabular views. The atlas and Trend views are both graphical views. The Summary view may contain both tabular and graphical elements.

There are many characteristics that are common to all views. This section describes the common tasks you can perform on the ZfS views, including:

- ◆ [“Changing the Appearance of a View” on page 54](#)
- ◆ [“Modifying Columns” on page 55](#)
- ◆ [“Filtering Views” on page 56](#)
- ◆ [“Sorting Views” on page 57](#)
- ◆ [“Printing a View” on page 57](#)
- ◆ [“Exporting a View” on page 58](#)
- ◆ [“Saving Views” on page 58](#)
- ◆ [“Deleting and Renaming Custom Views” on page 59](#)

Changing the Appearance of a View

In a view, you can change the following:

- ◆ [“Changing the Display Font” on page 54](#)
- ◆ [“Customizing Grid Lines” on page 55](#)
- ◆ [“Displaying the View Title” on page 55](#)

Changing the Display Font

To change the font of the text on a tabular view’s headings or rows:

- 1** Click View > Settings > Appearance.
The Appearance dialog box is displayed.
- 2** To change the header or row font, click the appropriate button as follows:
 - ◆ To change the header font, click the Header Font button.
 - ◆ To change the row font, click the Row Font button.The Fonts dialog box is displayed.
- 3** Select the font options you want > click OK to close the Fonts dialog box.
- 4** To save the changes made to the view, click View > Saving > Save.

Customizing Grid Lines

By default, the views displayed by ZfS do not contain grid lines. To display horizontal and/or vertical grid lines and to select a color for the grid lines:

- 1 Click View > Settings > Appearance.
The Appearance dialog box is displayed.
- 2 Select the grid line style you want to use from the Style drop-down list. You can choose to have:
 - ♦ No grid lines (default)
 - ♦ Horizontal grid lines only
 - ♦ Vertical grid lines only
 - ♦ Vertical and horizontal lines
- 3 If you want to select a color for the grid lines, click the Color button.
The Color Chooser dialog box is displayed. This dialog box includes three tab pages — Color Swatches, HSB, or RGB — allowing three methods of color selection.
- 4 Select the color you want to use for the grid lines using one of the three tab pages > click OK to close the Color Chooser dialog box.
- 5 Click OK to close the Appearance dialog box.
- 6 To save the changes made to the view, click View > Saving > Save.

Displaying the View Title

You may find it useful to display the view name at the top of the right frame to help you keep track of where you are within the ZfS ConsoleOne,

To display the view title:

- 1 Click View > Show View Title.

Modifying Columns

In a tabular view, you can change the columns in the following ways:

- ♦ [“Resizing Columns” on page 55](#)
- ♦ [“Adding and Removing Columns” on page 55](#)
- ♦ [“Changing the Column Order” on page 56](#)

Resizing Columns

To resize a column:

- 1 Move the mouse pointer to the margin between the columns you want to adjust.
- 2 When the pointer changes to a sizing arrow, drag the column to the width you want.
- 3 To save the changes made to the view, click View > Saving > Save.

Adding and Removing Columns

To add or remove columns from a view:

- 1** Click View > Settings > Column Selector.
- 2** To add a column, select the column name from the Available Fields list > click Add.
- 3** To remove a column, select the column name from the Show These Fields in This Order list > click Remove.
- 4** Click OK.
- 5** To save the changes made to the view, click View > Saving > Save.

Changing the Column Order

To change the order in which columns are displayed:

- 1** Click View > Settings > Column Selector.
- 2** Select the column you want to move from the Show These Fields in This Order list > click the Move Up or Move Down button to change the location of the column.
- 3** Click OK.
- 4** To save the changes made to the view, click View > Saving > Save.

Filtering Views

You can display the alarms in a tabular view based on filter conditions. The filter applies only to the current management session and clears once you exit ConsoleOne.

You set up a filter by selecting a criteria from four drop-down lists or entering a criteria. You can either set up simple filters that require only one line, or complex filters composed of multiple lines or groups of lines. If you set up a filter using more than one line, you must also specify the logical relationship between the line and/or group of lines.

To set up a filter:

- 1** Go to the required view.
- 2** Click View > Settings > Filter.
- 3** Select the column by which you want to filter alarms from the first drop-down list.
- 4** Select an operator from the second drop-down list.

The operator defines the constraint value set to the column. You can specify any of the following values for the alarm display - equal to, not equal to, greater than, less than, greater than or equal to, less than or equal to, contain, or start with the value you select in the third drop-down list. The list of available operators depends on the selected column.

- 5** Select a value from the third drop-down list.
- 6** Specify how this filter statement relates to other statements you plan to define by selecting a value from the fourth drop-down list.
 - ◆ If this is the only filter statement or if it is the last statement in a group, select End.
 - ◆ If you want to add a line below the current filter statement, select New Row. A new line is added. You must define the logical relationship between the previous line and the new line. The alarms will be displayed based on the logical condition you have specified. Select And to satisfy both the filter conditions. Select Or to satisfy any one of the filter conditions for the alarm to be displayed.

- ◆ If you want to add one or more lines that are unrelated to the preceding lines, select New Group. A new line is added. An additional drop-down list separates the new line from the preceding lines. Select a value from this drop-down list to indicate the relations between the filter statements. Select And if you want both the filter statements to be satisfied. Select Or if you want only one of the filter statements in one of the groups to be satisfied. Select End from the fourth drop-down list when you add a new group.

7 Click OK if you have finished defining filters.

The view is updated to display only those entries that meet the filter criteria you defined.

Sorting Views

Using the sorting feature to modify the order in which the entries in a tabular view. You can sort the entries in the following two ways:

- ◆ “[Sorting the View Using a Single Column](#)” on page 57
- ◆ “[Sorting the View Using Multiple Columns](#)” on page 57

For instructions on sorting alarms, see “[Sorting Alarms](#)” on page 112.

Sorting the View Using a Single Column

To sort the entries displayed in the view by a single column:

1 Double-click the column header for the column by which you want to sort the entries.

When you double-click the column header, the entries in the view are sorted by that column in descending order (the most recent entries first). To sort the entries by ascending order (oldest entries first), double-click the column header again.

Sorting the View Using Multiple Columns

To sort the view using multiple columns:

1 Click View > Settings > Sort.

2 Select the first column you want the entries sorted by from the Sort Items By field.

3 Select the appropriate radio button to indicate whether you want the entries sorted in ascending or descending order.

4 Select the second column by which you want entries sorted from the Then By field > click the ascending or descending radio button to specify the sort order.

5 Repeat [Step 4](#) for each subsequent column for which you want entries sorted.

6 Click OK.

The entries are now sorted according to the criteria you specified.

Printing a View

To print a view:

1 Go to the view you want to print.

2 Click File > Print.

3 In the Print dialog box, select the print options you want > click OK.

- 4 In the next Print dialog box, click OK.

Exporting a View

You can export a tabular or graphical view to one of the following file formats:

- ♦ HTML
- ♦ Comma-delimited text files (.CSV)
- ♦ Tab-delimited text files (.TXT)
- ♦ Blank-space-delimited text files (.TXT)

To export a view:

- 1 Go to the view you want to export.
- 2 Click File > Export.
- 3 From the Export File Type drop-down list, select the format to export the view.
- 4 Enter the path and name of the file you want to save in the Filename field or click Browse to search for a location you want to export the file to.
- 5 Click OK.

Saving Views

By default, any of the changes you make to the appearance, content, sorting, or filtering of a view are discarded when you exit ConsoleOne. If you want to retain the changes you have to explicitly save the view.

This section includes the following topics:

- ♦ [“Saving the Existing View” on page 58](#)
- ♦ [“Creating a New View” on page 58](#)
- ♦ [“Deleting and Renaming Custom Views” on page 59](#)

Saving the Existing View

If you want to permanently modify the existing view to reflect the changes you made, you can simply save the view as follows:

- 1 Modify the view as desired.
- 2 Click View > Saving > Save.

The next time you display the view, the changes will be retained.

Creating a New View

In some cases, you might find it useful to create a new view with the changes made. The existing view is left unmodified and you can save the new view under a different name.

To save the view under a new name:

- 1 Modify the view as desired.
- 2 Click View > Saving > Save As.

- 3** Enter a name for the view in the Enter New View Name field > click OK.

Deleting and Renaming Custom Views

To rename or delete the custom views you have saved:

- 1** Click View > Saving.
- 2** To rename a custom view, select the view from the Saved Views list > click Rename.
or
To delete a custom view, select the view from the Saved Views list > click Delete.
- 3** When you have finished modifying your saved views, click Close.

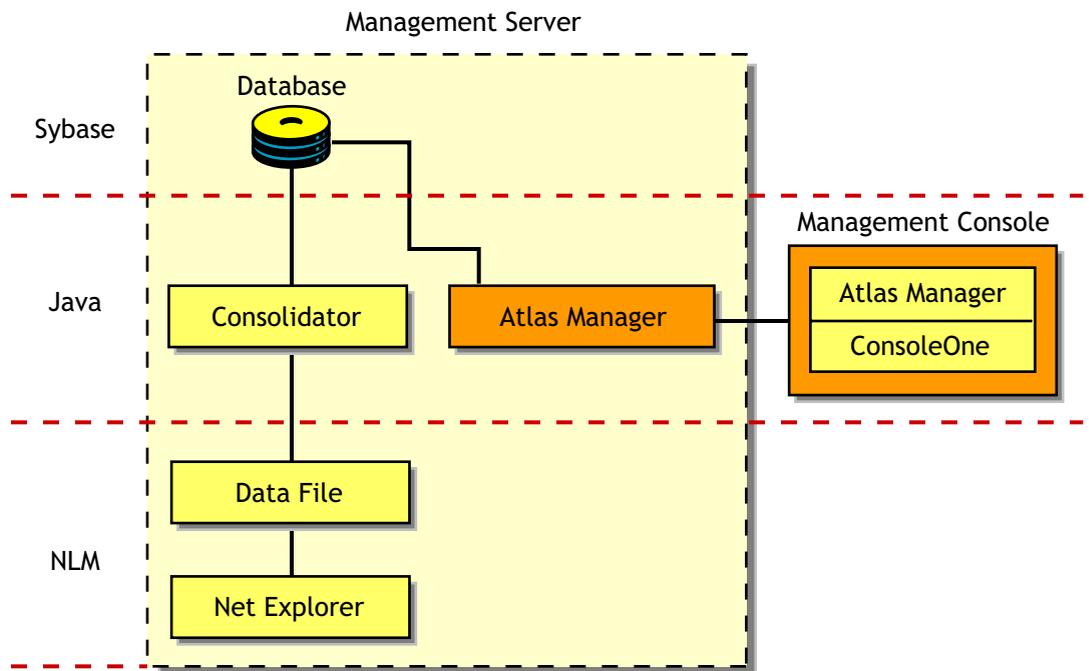
3

Understanding Network Discovery and Atlas Management

Discovery is the process of determining the topology of your network. You can manage, monitor and display the components of your network from ConsoleOne®. Discovery involves the following three major components of the ZENworks® for Servers (ZFS) software:

- ♦ **Discovery software:** A set of NetWare® Loadable Module™ (NLM™) files that run on a management server and discovers the network topology
- ♦ **Consolidator software:** Software that runs on the management server, which reads the data discovered by discovery, and populates the Topology database.
- ♦ **Atlas Manager software:** Software that reads the Topology database, creates an atlas database, and displays the network topology in an atlas on ConsoleOne.

The following figure shows a high-level view of the discovery components:



This section deals with the following topics:

- ♦ [“Understanding Network Discovery” on page 62](#)
- ♦ [“Setting Up Discovery” on page 88](#)
- ♦ [“Managing the Atlas” on page 99](#)

Understanding Network Discovery

The NetExplorer™ software drives the discovery process on the management server. The discovered information is populated in the Topology database. The Atlas Manager creates a related atlas database which encapsulates the topology information and adds information related to how the user views the maps.

The following sections will help you understand the network discovery process:

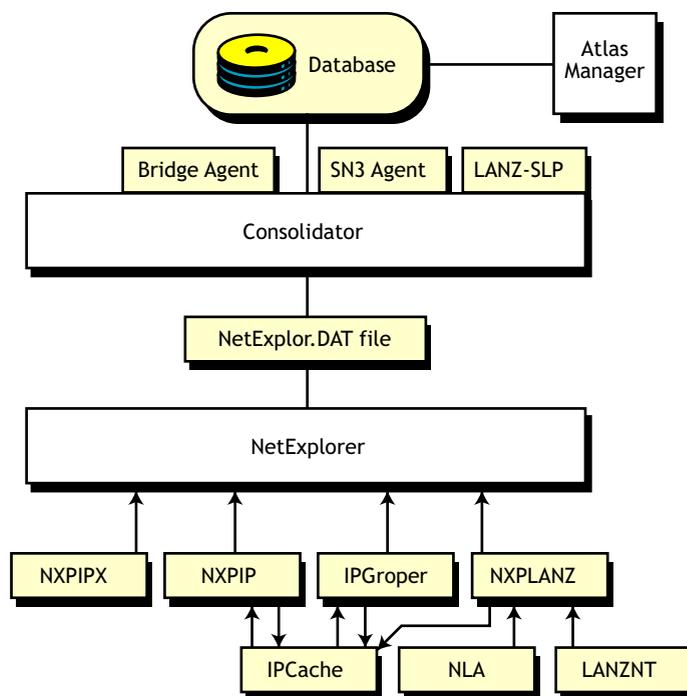
- ◆ [“Discovery Components” on page 62](#)
- ◆ [“Discovery Process” on page 68](#)
- ◆ [“What Is Discovered” on page 75](#)
- ◆ [“File-Based Discovery” on page 83](#)

Discovery Components

The NetExplorer and Consolidator software that runs on the management server aids in discovering your network and updating the database.

Your network is automatically discovered by NetExplorer when you start it for the first time.

The following illustration shows the discovery components on the server:



The NetExplorer system consists of the following interdependent components:

- ◆ [“Discovery” on page 63](#)
- ◆ [“Consolidator” on page 65](#)
- ◆ [“Atlas Manager” on page 66](#)
- ◆ [“Database Object Editor” on page 67](#)

- ◆ “Management Console Software” on page 67
- ◆ “Additional ZFS Components” on page 67

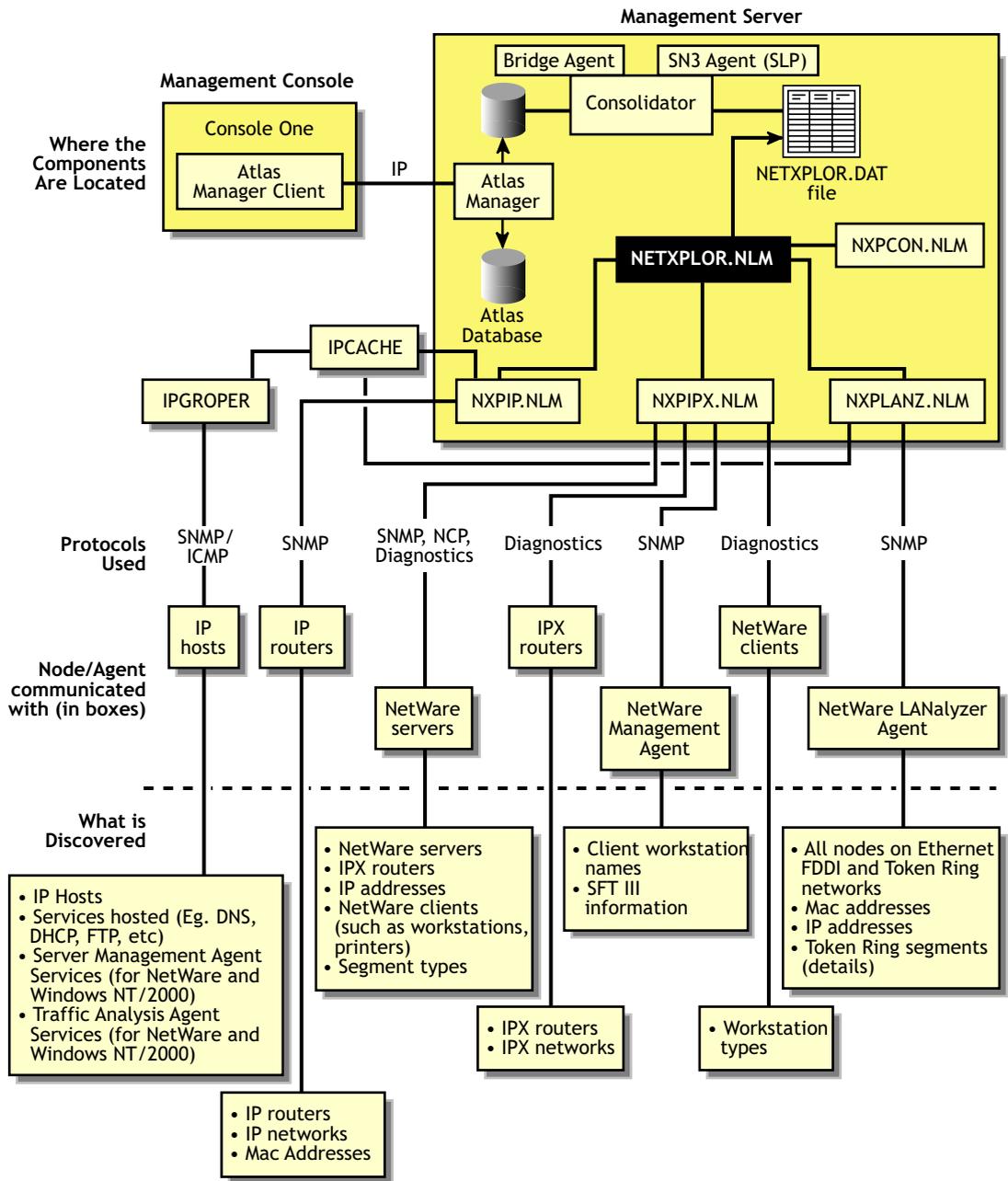
Discovery

The discovery software resides on the management server and uses the discovery NLM™ software to discover the various network devices.

- ◆ NXPIP.NLM discovers IP routers on IP networks and sends IP router information to discovery. It communicates with the IPCACHE module to share this information with IPGROPER.
- ◆ IPGROPER detects IP host addresses and the following services: Domain Name System (DNS) names, Dynamic Host Configuration Protocol (DHCP) services, Telnet, Hypertext Transfer Protocol (HTTP), Simple Mail Transfer Protocol (SMTP), and File Transfer Protocol (FTP).
- ◆ NXPIX.NLM discovers various NetWare systems on IPX™ networks and sends information about systems to NetExplorer.
- ◆ NXPLANZ.NLM communicates with Traffic Analysis Agents (LANalyzer®) for NetWare and Windows* NT* to gather information about all systems communicating on the segments that are monitored, and sends this information to discovery.

The following figure illustrates the architecture of the discovery system and shows the roles of the various components, network systems, and agent software.

IMPORTANT: Discovery uses the server and traffic management agents to obtain certain discovery information. Though not required, using these agents across your network enhances the accuracy and detail of logical maps displayed by ConsoleOne.



Supported Protocols

ZfS software supports the Service Location Protocol (SLP) on NetWare 5.x networks to enhance the discovery speed.

The server management and Traffic Analysis Agents for NetWare use the Service Advertising Protocol (SAP) to identify themselves to other components. SAP filtering prevents routers from passing SAP packets. To enable the management server and ConsoleOne to receive the SAP packets that identify manageable servers, Hub Management Interface (HMI) hubs, and other servers, configure the router that is filtering SAP packets to list the specific SAP numbers that it should pass. NetWare systems and ZfS components use the SAP numbers listed in the following table.

Component	SAP Number (Decimal)	SAP Number (Hexadecimal)
NetExplorer NLM	567	237
NetWare Management Agent	635	27B
ManageWise Agent for Windows NT server	651	28B
NetWare LANalyzer [®] Agent [™] (Traffic Analysis Agent)	570	23A
Print server	7	7
NetWare file server	4	4

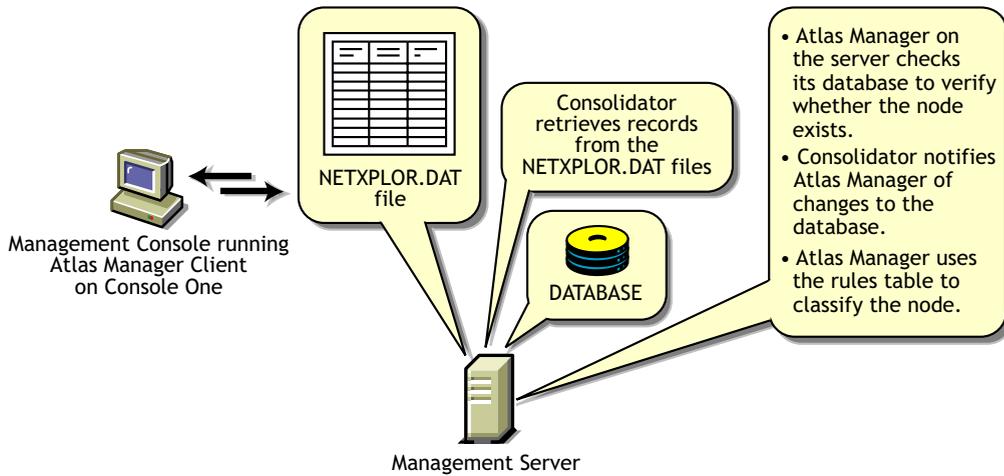
Consolidator

The Consolidator software resides on the management server and performs the following tasks:

- ◆ Reads the NetExplorer data files, which contains all the discovered information.
- ◆ Interprets the records in the NETXPLOER.DAT file.
- ◆ Checks whether the system has already been created in the Topology database. If the system does not exist in the Topology database, the Consolidator creates the system.
- ◆ Uses the Bridge agent to query the Bridge Management Information Base on IP networks and discovers which systems are connected to a port of a bridge.
- ◆ Uses the SN3 agent to get the eDirectory name of NetWare servers. The SN3 agent enhances the performance of discovery by using SLP to discover NetWare 5.x servers.
- ◆ Runs the MIBCOMPILER.RULE file on all the discovered devices and verifies for the MIBs mentioned in the rule file on these devices and updates the database. You can also add or delete the MIBs in the MIBCompiler.rule.
- ◆ Writes discovery information to the ZfS database.

The following figure shows the tasks of the Consolidator. NETXPLOER.NLM creates the NETXPLOER.DAT file and the Consolidator starts reading the records from the file. If NetExplorer processes are restarted, the NETXPLOER.DAT file is re-created and the Consolidator requests the first record in the new file.

When the Consolidator retrieves a record from the NETXPLOER.DAT file, it searches for the record in the database. If the system is not in the database, the Consolidator inserts it and notifies the Atlas Manager of the update.



Command Line Options

If you want to manually operate the Consolidator, use the command line options shown in the following table.

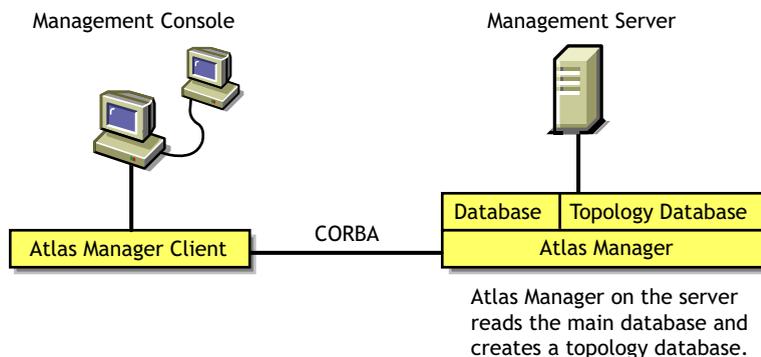
Option	Allows the Consolidator to
-notify	Notify the Atlas Manager that it has updated the database.
-database <i>data_path</i>	The specific location of the database file to perform operations on.

Atlas Manager

The Atlas Manager software consists of a server and a client component. The server component resides on the management server along with the ZfS topology database. The Atlas Manager server component retrieves discovery data from the topology database and creates its own atlas database.

The client component of the Atlas Manager resides on ConsoleOne. The server component can communicate with several console components at any given time. Any changes made to the maps on the console (for example, rename, import, and layout) are communicated to the Atlas Manager server component, to update the atlas database.

The following figure shows the Atlas Manager server and client software:



The Atlas Manager looks for a rule in its rules table to help classify the system. The rules help the Atlas Manager make decisions, such as which icon should be used to display the system on the

maps. If the system in the record matches one of the rules, the Atlas Manager updates the database according to the rule.

Database Object Editor

The Database Object Editor supplements the discovery system. Sometimes discovery might not discover devices on your network, or might display incorrect information of the devices on your network. You can use Database Object Editor to add the missing entities into the database or edit incorrect information of the entities.

The Database Object Editor client uses ConsoleOne snap-in to display the user interface. Using the Database Object Editor, you can perform operations on a segment or a node.

The Database Object Editor Server interacts with the Consolidator to process information related to the node and segment object and populates the topology database with this information.

You can use the Database Object Editor to add or delete a segment or a node and modify the segment or the node information.

To add a segment or a node:

- 1 From ConsoleOne, select Tools > Database Object Editor > New.
- 2 Enter the details for the segment or the node.
- 3 Click OK.

To edit the information about the segment or the node:

- 1 From ConsoleOne, select the segment or the node you want to edit.
- 2 Select Tools > Database Object Editor > Edit.
Modify the required information.
- 3 Click OK.

To delete the segment or the node:

- 1 From ConsoleOne, select the segment or the node you want to delete.
- 2 Select Tools > Database Object Editor > Delete.

Management Console Software

The management console software snaps in to ConsoleOne. Management sites are created in ConsoleOne. In each site, an atlas is created that maintains the integrity of the discovery information.

Additional ZfS Components

NXPIP.NLM, NXPIPX.NLM, and NXPLANZ.NLM operate in conjunction with the following components:

- ◆ [“Traffic Analysis Agent for NetWare Servers” on page 68](#)
- ◆ [“Server Management Agent for NetWare Servers” on page 68](#)
- ◆ [“Bindery of NetWare Servers” on page 68](#)

Traffic Analysis Agent for NetWare Servers

The traffic analysis (LANalyzer) agent for NetWare is a set of NLM files that provides traffic analysis of Ethernet, Fiber Distributed Data Interface (FDDI), or token ring segments. The Traffic Analysis Agent discovers all systems on the segments it monitors, regardless of the protocols the systems use. You can monitor multiple segments by placing agents on each segment.

The NXPLANZ.NLM software on the management server uses SNMP to query servers running the Traffic Analysis Agent for information about each system that resides on their segments.

IMPORTANT: For an effective discovery process, you should have the Traffic Analysis Agent monitoring each source-routed token ring segment.

Server Management Agent for NetWare Servers

To discover IPX servers and workstations, managed servers are any NetWare 4.x, NetWare 5.x, or NetWare 6 servers with the server management agent installed. Server management agents respond to SNMP queries from NXPIPX.NLM with the username and address of those workstations that are logged in to the server. NXPIPX.NLM obtains SFT III™ server information from the server management agent. For effective results, you should install a management agent on every NetWare 3.x, NetWare 4.x, or NetWare 5.x, or NetWare 6 server on your network.

Bindery of NetWare Servers

NXPIPX.NLM queries all NetWare servers for information in their binderies. All NetWare servers allow their binderies to be examined by the discovery process when their security settings are set to the default values.

For the NetExplorer NLM software to discover the login names of workstations attached to a Netware server, a server management agent must be installed on the server.

Discovery Process

NetExplorer discovers your network continually. The following sections discuss the discovery processes:

- ◆ [“Discovery Cycles” on page 68](#)
- ◆ [“Continuous Discovery” on page 73](#)

Discovery Cycles

When you first start discovery, you should let it run as long as necessary to build the baseline data. Very small networks might take one or two hours, while very large networks (several thousand nodes) might require a day or two to be discovered.

The discovery process occurs in cycles. A cycle is the process by which a discovery module identifies every node it can at a time. You can configure discovery on the server to discover only certain addresses, thus reducing the duration of a cycle. For more information, see [“Changing the Discovery Scope” on page 93](#).

The initial cycle continues until no additional devices are discovered. This initial cycle gathers information that might be insufficient to classify certain devices or to identify the correct segment for each device. Further discovery cycles provide additional, new, and changed information. As discovery cycles proceed, the information becomes more accurate.

Each discovery process queries the network using different methods to discover systems. Four independent discovery modules run in the order mentioned below during each discovery cycle:

1. IP router discovery on IP networks only.

This process, run by the NXPIP module, starts from the local router. Using the local router's routing table information, NXPIP discovers other routers on the network. It then uses the routing table information to further discover the network. This process is repeated for each router discovered.

The NXPIP module stores the router address information and information about any IP-bound network device in the IPCACHE module.

NXPIP.NLM is installed on the management server. It uses SNMP to discover IP routers. To use this NLM, your management server must also be running TCP/IP bound to at least one of your network's interface boards. NXPIP.NLM uses MIB-II information, such as the system table, routing table, interface table, interface data-link type and frame type, and segment data-link type. Note that because there are different versions of MIB-II implementations for different vendors, the information you receive might differ.

IMPORTANT: If you have specified an additional level of control by allowing certain IP addresses to perform SNMP queries to the routers, ensure that the IP address given to the ZfS server is privileged to query all the routers in the network. Otherwise, discovery will not be complete, and incomplete network information will appear in the Islands page of the atlas.

2. IP discovery of workstations and servers.

This process, run by the IPGROPER module, receives the router and network information written into the IPCACHE by the NXPIP module as the input. RMON, based discovery run by the NXPLANZ module also writes the information about the networks and IP hosts that it discovers into IPCACHE. This also acts as an input to the IPGROPER module.

It queries each router that has been discovered by NXPIP for its ARP tables, identifying each active IP host on the network. For IP addresses that are not found in the ARP table of any of the routers, IPGROPER tries to ping and identify whether a host by that IP address is alive.

IPGROPER queries each IP host that is identified to be alive for information about the following hosted services: HTTP, DHCP, Telnet, SMTP, and DNS. It also verifies whether the server management software and the Traffic Analysis Agents are installed and running on this host.

Simultaneously, the IPGroper module queries the DNS server specified in the SYS:\ECT\RESOLV.CFG file on the management server for the DNS names of all these IP hosts.

IMPORTANT: For a server or a segment to be manageable, it is important to discover the server management agent and the Traffic Analysis Agent running on an IP host on that server or the segment.

3. IPX discovery on all networks, including NetWare/IP networks:

This process, run by the NXPIPX module, starts at the management server itself to discover its IPX address, the LAN type of each adapter, and SAP information about other known devices and their services. After gathering this information, NXPIPX requests the same types of information from each device listed in the bindery. This process is repeated each time NXPIPX discovers a new device.

NXPIPX.NLM uses a variety of NetWare, SNMP, and IPX protocols, such as IPX diagnostics, to discover NetWare servers, IPX routers, and IPX workstations.

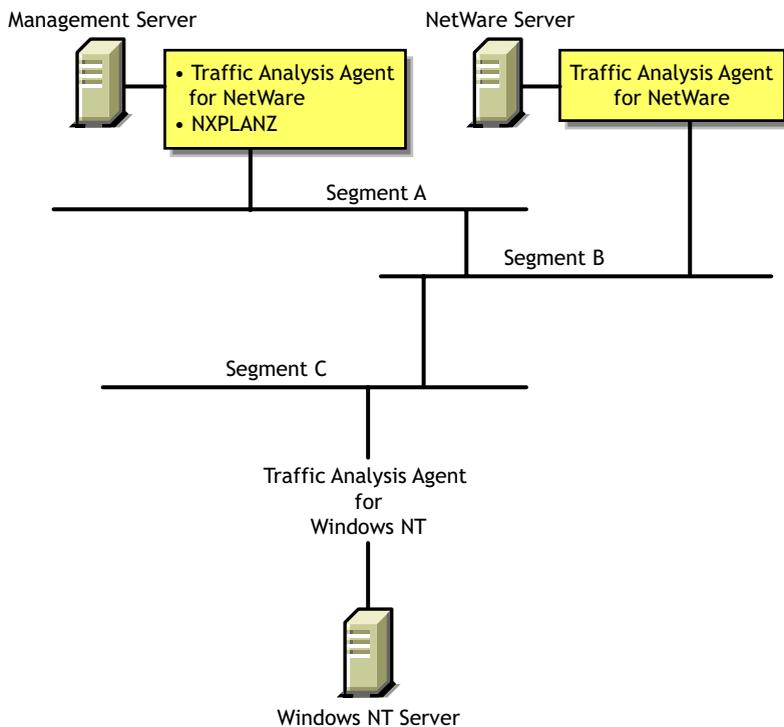
IMPORTANT: When NXPIPX.NLM is loaded, a working directory named NXPWORK is created by default under the *install_volume\install_dir\ZENWorks\MMS\MWSERVER\NMDISK* subdirectory. During installation, you can specify a different path to create the NXPWORK subdirectory. NXPIPX puts all of its temporary files in this directory. Do not read, modify, or delete any file in this directory because this might cause some discovery process to not function.

4. RMON based discovery of IP Hosts.

This process, run by the NXPLANZ module, starts by identifying all the remote agents, which includes the Traffic Analysis (LANalyzer) Agents for NetWare and Windows NT. The Traffic Analysis Agents on a segment discover devices based on the IP address to MAC address binding data contained in packets that are transmitted on the segment. The NXPLANZ module on the management server retrieves the data by using SNMP to communicate with the Traffic Analysis Agents.

The NXPLANZ module reports information about the LANalyzer agents on your network and the IP hosts on the segments monitored by these LANalyzer agents to NetExplorer. The information about the networks monitored by the LANalyzer agents and IP hosts on the monitored networks is also written to IPCACHE to enhance the effectiveness of service discovery by the IPGROPER module.

The following figure shows NXPLANZ querying Traffic Analysis Agents software on segments B and C, respectively.



To improve the effectiveness of the discovery, ensure that the LANalyzer agent is installed and running on each network segment that you want to discover. If SLP is disabled on your network or if SAP packets are filtered by the routers in your network, NXPLANZ may not be able to discover all the LANalyzer agents in the network.

In order to ensure that all the LANalyzer agents on your network are being queried by the NXPLANZ module, specify these LANalyzer agents explicitly using NXPCON.

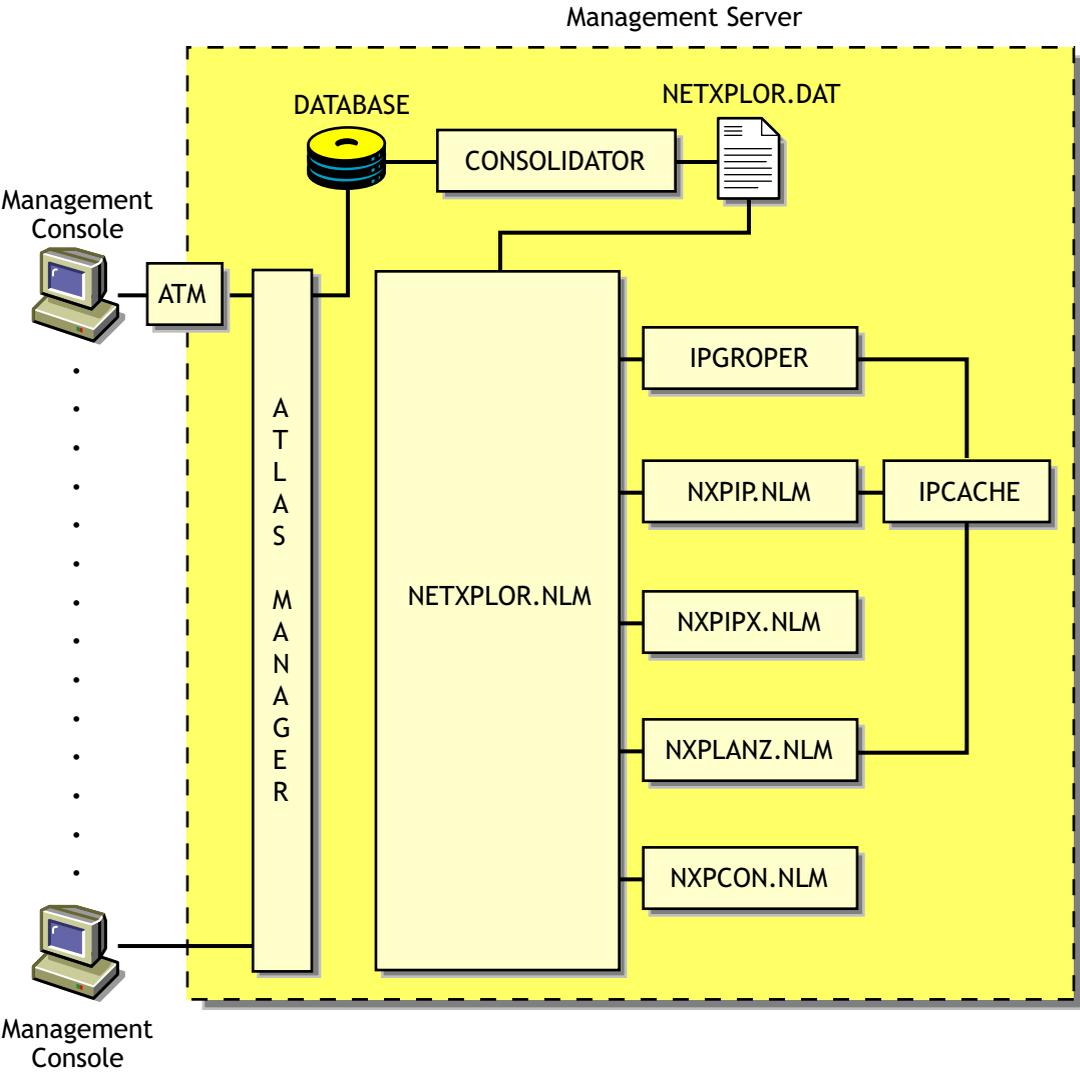
During the initial discovery cycle, these modules run sequentially. As a result, information about the Traffic Analysis Agent software is discovered late.

In later discovery cycles, the four modules run concurrently. They continue their discovery processes, but send only new or changed data to NETXPLOE.NLM. As additional data arrives,

segments can be consolidated, devices can be placed on the appropriate segments, and new devices can be discovered.

Each succeeding cycle of different discovery NLM files has the potential to provide key information that finally identifies a device and provides sufficient data for NetExplorer to consolidate the data.

The data discovered by the NLM processes is communicated to ConsoleOne through the Atlas Manager. The following figure shows the relationship of the discovery NLM processes, NetExplorer, and ConsoleOne. See [“Discovery Process” on page 68](#) for a description of how these pieces operate together to discover the contents and topology of a network.



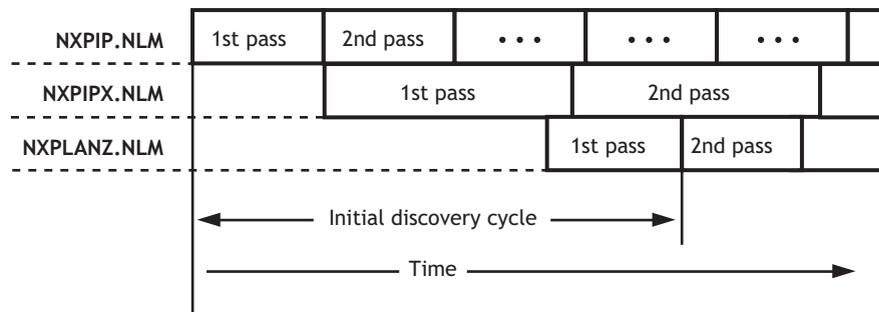
The following table summarizes the default seed and scope and user-definable changes for each discovery module:

Discovery Module	Default Seed Information	Default Scope	User-Definable Changes
NXPIP	Examines the management server routing table. Places the router addresses in the IPCACHE module.	Entire network if community string matches.	Reduce scope by specifying IP scope information in NXPCON. If public SNMP community string is not used, list SNMP community strings of routers in NXPCON.
IPCACHE	Supporting module in NetExplorer. Contains temporary information about devices and networks which is used by NXPIP, IPGROPER and NXPLANZ.		
IPGROPER	<ol style="list-style-type: none"> 1. Queries each router address in IPCACHE for ARP tables to identify network devices. 2. Queries each network device for the services it hosts (FTP, HTTP, Telnet, SMTP, DNS, and DHCP) and their DNS names. 3. Discovers hosts running server management and Traffic Analysis Agents. 	All IP networks connected to routers already discovered by NXPIP	<ul style="list-style-type: none"> ◆ Enable or disable autodiscovery ◆ Enable or disable file-based discovery
NXPIPX	Examines the management server's configuration.	Entire IPX internetwork.	Reduce scope by specifying IPX scope information in NXPCON.
NXPLANZ	Examines the list of servers running Traffic Analysis Agent software listed in NXPCON.	All segments with Traffic Analysis Agent software.	Specify name and IP addresses of Traffic Analysis Agent for Windows NT in NXPCON. If SLP is disabled or SAP is being filtered, specify the name and address in NXPCON for the Traffic Analysis Agent for NetWare.

Continuous Discovery

NetExplorer discovers the internetwork on which it resides, through a process initiated and controlled by NETXPLOR.NLM. Initially, each discovery NLM identifies itself to NETXPLOR.NLM, which then begins the initial discovery cycle. The cycle starts with NXPIP discovery, followed by NXPIPX discovery, and finally NXPLANZ discovery. The discovery cycles of IPGROPER are not controlled by NETXPLOR.NLM. Once started it runs continuously. Information gathered by NetExplorer is stored in the NETXPLOR.DAT file on the management server.

In the following figure, each of the discovery processes is shown in relationship to time. Once NXPIP finishes its first pass, NXPIPX begins and NXPIP starts over. After NXPIPX finishes its first pass, NXPLANZ begins and NXPIPX starts its second pass. Unless otherwise directed, all three of the discovery processes run continually to detect changes to the network. Any changes to the network are saved as records in the NETXPLOR.DAT file. When all three discovery processes have completed one pass, the initial discovery cycle is complete.



The following sections describe each sequence in greater detail:

- ◆ [“NXPIP” on page 73](#)
- ◆ [“NXPIPX” on page 73](#)
- ◆ [“NXPLANZ” on page 74](#)
- ◆ [“NETXPLOR” on page 74](#)
- ◆ [“SNMP Community String Discovery” on page 75](#)

NXPIP

The first sequence in the NetExplorer discovery cycle involves the discovery of IP routers. NXPIP locates its local router using TCP/IP configuration information. NXPIP then queries the router for the identity of other routers on the network. NXPIP queries the MIBs on the routers using SNMP to collect the IP addresses, interface types, and MAC addresses.

By default, NXPIP attempts to discover your entire IP network. You can restrict the scope of the IP discovery by specifying the scoping information in NXPCON.

NXPIPX

NXPIPX uses a series of techniques, including SNMP, RIP, IPX, and SPX™ diagnostics to discover the attached IPX or NetWare/IP internetwork. After NXPIP completes its first pass, NXPIPX begins discovery at the management server. NXPIPX examines its own server and discovers the names of other servers. It then queries each of these servers to discover more servers and repeats this process until no more servers are found.

SNMP Community String Discovery

Each time NetExplorer tries to access a system through SNMP, it uses the community strings that have been configured using the NXPCON utility on the management server. When it encounters a new system, it tries each of the configured community strings. After it has found a community string for a particular IP or IPX address, it records this name in a file so that in subsequent cycles it does not need to retry with the other configured names.

You can view these community strings using NXPCON. The community strings are used in the order specified. Therefore, the most-used community string should be configured first in the list.

IMPORTANT: An SNMP query with an invalid SNMP community string results in no response from the target system and the request times out.

What Is Discovered

NXPIP, NXPIPX, and NXPLANZ use a variety of techniques to discover the following categories of network objects and present them in the atlas:

- ◆ “Systems” on page 75
- ◆ “Network Segments” on page 81

Generally, information gathered by NXPIP and NXPIPX is sufficient to place systems on the network maps correctly. When NXPIP and NXPIPX have not discovered systems, NXPLANZ retrieves MAC addresses collected by the Traffic Analysis Agent software and the new systems are added to the database. Consequently, all systems are discovered on segments monitored by the Traffic Analysis Agents.

Systems

The following table shows the different types of systems discovered:

System	Comment
NetWare Management Agent	Service type of 563 decimal (NetWare Management Agent 1.5 or 1.6) or 635 decimal (NetWare Management Agent 2.6) or NetWare Management Agent MIB implemented.
Management Agent for Windows NT/2000	NT Management Agent MIB implemented.
NetWare LANalyzer Agent	Service type of 570 decimal or LANalyzer MIB implemented.
LANalyzer Agent for Windows NT/2000	LANalyzer MIB implemented
NetWare File Server	Service type of 4 (file server). NXPIPX discovers all NetWare 3.x, 4.x, and 5.x servers.
NetWare Print Server™	Service type of 71 or 7 decimal.
IPX Router	System with more than one adapter connected to different IPX networks.
IP Router	System that is configured as an IP router in MIB-II (IP forwarding enabled).

System	Comment
NetWare Client Workstation	System that responds to IPX diagnostics requests as an IPX workstation (has the NetWare Shell loaded).
SFT III IOEngine	Discovered by the IPX discovery module; responds with diagnostic information.
SFT III MSEngine	Discovered by the IPX discovery module.
Network Printers	Discovered if the printer generates a well-known service type.
NetWare Connect™	Service type of 590 decimal.
NetWare Communications Server	Used by the NetWare for SAA* services manager products; has a service type of 304 decimal.
Management Server	Running discovery NLM files; has a service type of 567 decimal.
Any System	Any system is discovered if it is connected to a LAN segment being monitored by a Traffic Analysis Agent.

The different types of services discovered are Telnet, HTTP, DNS, SMTP, DHCP, Routers, eDirectory, SFTIII, and SNMP.

The following sections contain more information about the various systems that are discovered:

- ◆ “NetWare Client Workstations” on page 76
- ◆ “IP Routers” on page 77
- ◆ “NetWare SFT III Servers” on page 78
- ◆ “Systems Not Equipped with the IPX Diagnostic Responder” on page 79
- ◆ “Routers that Use Duplicate MAC Addresses” on page 79
- ◆ “Third-Party Routers” on page 79
- ◆ “NetWare MultiProtocol Router with WAN Ports” on page 80
- ◆ “IPX Networks” on page 80
- ◆ “IP Networks” on page 80
- ◆ “On-Demand Links” on page 80
- ◆ “Third-Party Routers with WAN Ports” on page 81
- ◆ “NetWare Connect Servers” on page 81
- ◆ “Virtual Switches” on page 81

NetWare Client Workstations

NXPIPX discovers all NetWare client software attached to discovered NetWare 3.x, 4.x, and 5.x servers. Clients that are turned off or are not attached to a server are not discovered. For this reason, a NetExplorer process that is run at night or on a weekend might not yield a complete map. Note that NetWare clients must have IPX diagnostics enabled.

When you configure a NetWare client to perform a bindery login, consider the scenarios in the following table:

Server	Bindery Login—What Is Discovered
NetWare 3.x, 4.x, or 5.x with server management agent installed	Workstation discovered; name is discovered only if logged in with IPX as the transport for NetWare 4.x and NetWare 5.x
NetWare 3.x, 4.x, or 5.x	Workstation discovered; name is not discovered

When you configure the client to perform a directory login, NetExplorer discovers only those systems that are *logged in* to an eDirectory tree and not those that are merely *attached* to the eDirectory tree. NXPIPX uses SNMP community string to communicate with the management agent and query on all NetWare servers for the username.

After NetExplorer discovers a NetWare client, NXPIPX queries the client using the IPX diagnostic protocol to confirm the discovery and gather more information about it. If IPX diagnostics are turned off, NXPIPX does not report the system. This applies to printers as well.

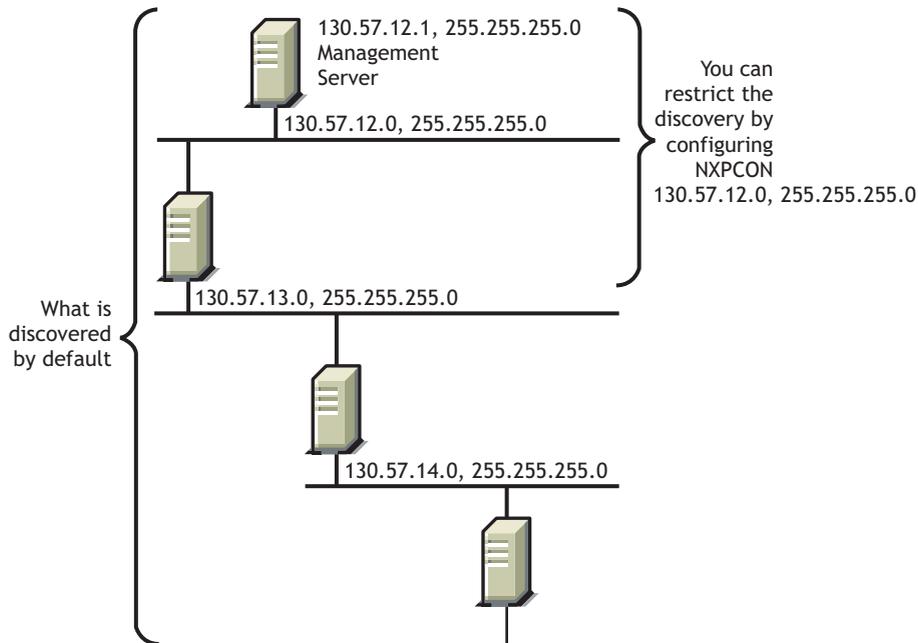
IP Routers

NXPIP uses SNMP to query all IP routers on the network by using the SNMP community string used by the routers. You must enter the list of community strings used by your routers using NXPCON.

You can configure this information into the router's MIB by using any SNMP configuration tool, including the SNMP MIB browser. If you configure router information such as the system name in the routers SNMP MIB, the discovery process records it in the database, allowing IP routers to be displayed with meaningful names.

By default, IP discovery discovers the entire network. The exploration can be restricted by specifying network numbers using the NXPCON Discovery Scope > IP Discovery Scope option. Also, if there are redundant IP routers, use the NXPCON IP Discovery > IP Routers option to specify the redundant IP router address; otherwise, NXPIP does not discover it. As shown in the following illustration, if the management server IP address is 130.57.12.0, the IP discovery NLM discovers the entire 10.57.85.0 network and its subnets.

The following figure shows how ZfS discovers IP routers:

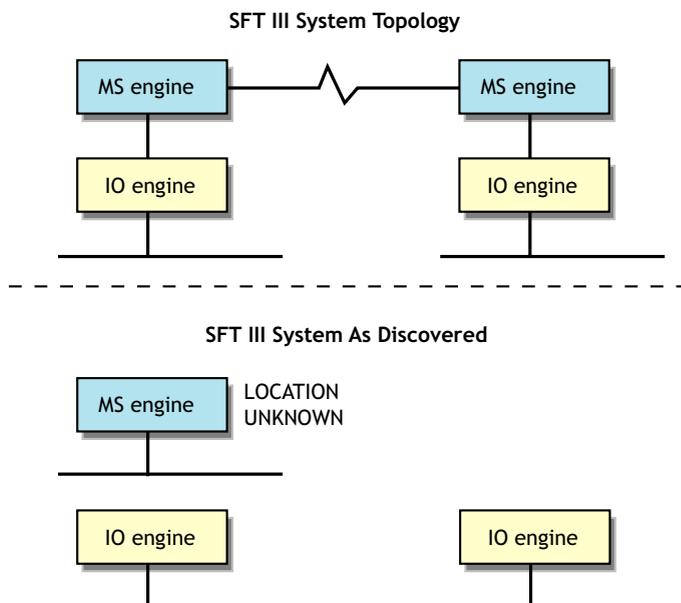


NetWare SFT III Servers

A NetWare SFT III™ server usually consists of two computer systems, each containing an input/output engine (IO engine) and a mirrored server engine (MS engine). Therefore, physically there are two IO engines and two MS engines; logically there are two IO engines and one MS engine.

If NetWare Management Agent is loaded on an SFT III server, the MS engine and both IO engines are discovered correctly with their names and placed in the correct segment in the atlas. However, the MS engine is placed in the Islands page. This happens because the two MS engines are associated with only one logical server on the network, and the location of the MS engine might change depending on which copy of the MS engine is the primary at any given time.

The following figure illustrates NetWare SFT III server discovery:



If the server management agent is not loaded on the MS engine, Discovery discovers only the MS engine and the IO engine that are primary at the time of discovery. The primary IO engine is labeled Noname in the area page. To change the name of an IO engine on a segment map, right-click the icon and click Rename.

Systems Not Equipped with the IPX Diagnostic Responder

NXPIPX discovers the following systems, but does not necessarily place them correctly in the atlas:

- ◆ NetWare for UNIX* servers
- ◆ Portable NetWare servers
- ◆ Access servers
- ◆ Modem servers
- ◆ Print servers

Because these systems do not respond to IPX diagnostics, they cannot answer queries from NXPIPX. Consequently, the LAN information required to place them on the maps might not be available. In this situation, NetExplorer places these systems in the Islands page of the atlas. In most cases, the presence of a Traffic Analysis Agent on each segment on which these systems appear, enables NetExplorer to obtain the missing information and correctly locate the systems in the maps.

If these systems are running IP, they will be discovered and placed correctly in the maps.

Routers that Use Duplicate MAC Addresses

NetExplorer can experience difficulties in discovering some routers because of the method routers use to identify their adapters. In some cases, the same MAC address is used on several network interfaces of a router. In these cases, it appears to NetExplorer that one adapter is connected to multiple segments. Unless otherwise specified, NetExplorer interprets multiple adapters as one adapter.

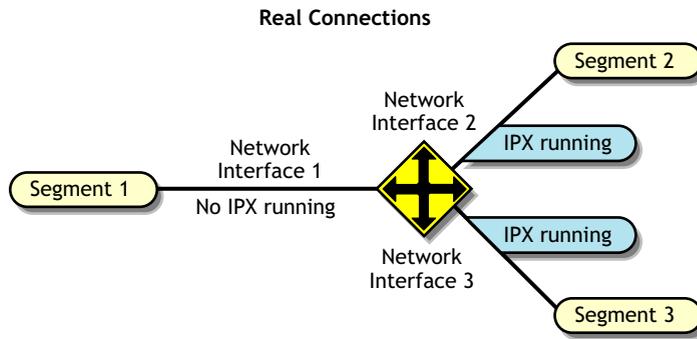
The multiple segments connected to the adapters are seen as one segment and NetExplorer consolidates the multiple segments.

Third-Party Routers

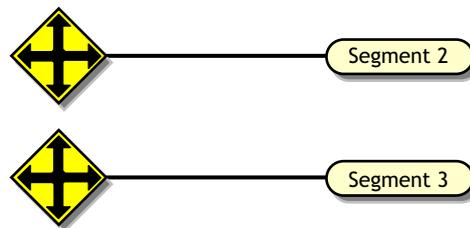
NXPIP discovers IP-bound interfaces only. When IP is not running on a router, NetExplorer discovers the IPX-bound interfaces, which results in:

- ◆ A separate router icon is shown for each interface in the router.
- ◆ Discovered interfaces are not placed in the same router in the atlas. Therefore interconnections are incorrect on the internetwork map and the router appears as separate, multiple routers, each containing one network interface from the real router.

The following diagram illustrates a router with IPX running on Network Interfaces 2 and 3 but not on Network Interface 1. NetExplorer places this router on the internetwork map as two separate systems. As shown, the connection to Segment 1 is not displayed, and the connections to Segments 2 and 3 are shown attached to two separate systems.



ZENworks for Servers Internetwork Map Connections



NetWare MultiProtocol Router with WAN Ports

NetWare MultiProtocol Router™ (MPR) 3.0 is now bundled with NetWare 5.x.

IPX Networks

NetWare MPR™ 3.0 reports the correct segment type of the WAN links. NetExplorer detects these correctly and displays them with the appropriate icon.

IPXWAN links between NetWare MPR 3.0 systems do not have an IPX network associated with them. When NetExplorer discovers such a link, it creates a name for the WAN segment of the form #UNNUM -*n*, where *n* is an integer assigned to make the segment name unique. On multi-access networks, such as frame relay and X.25, each connection in the network adds another #UNNUM -*n* to the segment name.

IP Networks

With NetWare MPR 3.0, you can configure both numbered and unnumbered IP links. NetExplorer discovers numbered links correctly. NetExplorer does not discover unnumbered IP links, resulting in the Islands page.

If IP is running on a third-party router and NXPIP is running on the management server, NetExplorer discovers only the IP-bound interfaces. The router is shown correctly in the atlas. If IP is not running on a third-party router but NXPIPX is running on the management server, NetExplorer discovers the IPX-bound interfaces. However, these IPX-bound interfaces are not placed in the same router icon in the atlas.

On-Demand Links

An on-demand link is a WAN connection between two routers in which only user data (no routing traffic) is exchanged across the link. The link is brought up only when there is data to send.

NetExplorer discovers on-demand IP and IPX links correctly, if sufficient static routing information has been configured to allow the management server to reach the other side of the on-demand link.

However, if a link is an on-demand and unnumbered IP link, the entire topology on the remote end of the link is not discovered. Click IP Discovery > Additional IP Routers in the NXPCON utility to configure an additional IP router address for the missing router.

Third-Party Routers with WAN Ports

NetExplorer discovers third-party routers correctly if they support MIB-II SNMP. Certain third-party routers can have a WAN link with no IP or IPX network number on the link. In this case, the WAN link is not discovered.

NetWare Connect Servers

NetExplorer discovers NetWare Connect servers; however, if you have more than one NetWare Connect[®] server on the network, NetExplorer consolidates them and they appear as one server.

Virtual Switches

A virtual switch is represented by the same icon used for a switch or bridge in the atlas maps. The display name of a virtual switch is always shown as the "switch on *IP address of network*." It is primarily used in atlas maps to display a meaningful network topology when discovery information is incomplete.

A virtual switch is shown in atlas maps under the following conditions:

- ◆ When two or more different physical media are connected by a switch, but the switch is not yet discovered. The virtual switch will disappear as soon as the real switch is discovered.
- ◆ When two or more different physical media are connected by a switch, the switch is configured with SNMP community strings other than public, and the SNMP community strings of the switch were not provided through NXPCON before starting discovery.
- ◆ When two or more different physical media are connected by a non-manageable switch or a hub.

Network Segments

NetExplorer discovers the following network segments:

- ◆ [“LAN and WAN Segment Types” on page 81](#)
- ◆ [“Source-Route Bridged Token Rings” on page 82](#)

NetExplorer cannot fully discover the following:

- ◆ [“Transparent Bridges” on page 83](#)
- ◆ [“Configuration Changes” on page 83](#)

LAN and WAN Segment Types

NetExplorer discovers the LAN and WAN segment types shown in the following table:

Known Segments in CIM Database	Unknown Segments in CIM Database
ATM	LAN: ARCnet
LAN: FDDI	LAN: LocalTalk*
LAN: Ethernet	SMDS
LAN: Token Ring	WAN: ISDN
WAN: X.25	WAN: SDLC
WAN: PPP	WAN: Serial
WAN: Frame_Relay	WAN: T1
	WAN: T3

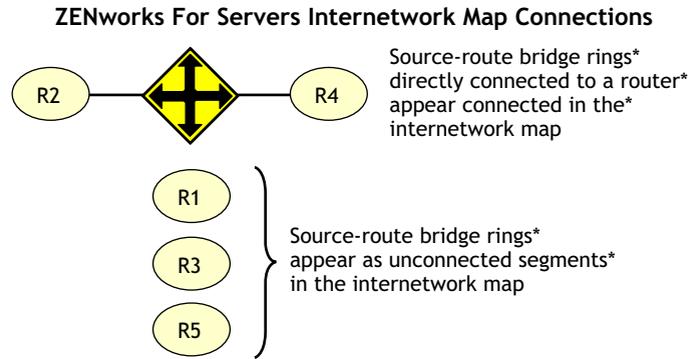
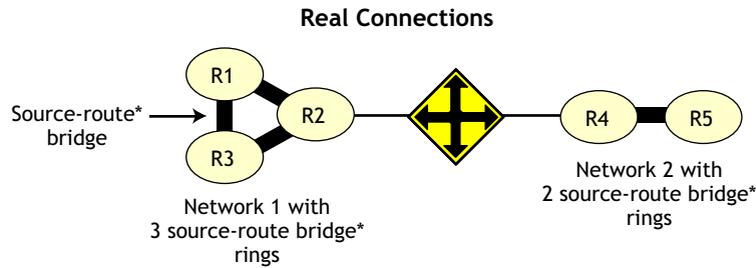
These values are discovered correctly if a system connected to the segment responds with an interface type from MIB-II RFC 1573.

Source-Route Bridged Token Rings

Atlas Manager displays source-route bridged token rings depending on whether the Traffic Analysis Agent for NetWare is installed on each ring.

- ◆ If you do not have the Traffic Analysis Agent for NetWare installed on each source-route bridged token ring in your network, NetExplorer discovers the network but consolidates all source-route bridged token rings that share the same IPX network number or IP subnet into a single segment. For example, in the following figure, rings R1, R2, and R3 are displayed as one segment, and rings R4 and R5 are displayed as another segment on the internetwork map.
- ◆ If you have the Traffic Analysis Agent for NetWare installed on each source-route bridged token ring, each Traffic Analysis Agent for NetWare discovers its own ring (segment) and every system on it. Atlas Manager displays the ring as a disconnected segment on the internetwork map.
- ◆ If you have the Traffic Analysis Agent for NetWare installed on a source-route bridged token ring connected to a router, the WAN page in the atlas shows the correct connections. However, if two networks each have several rings and only one ring in each network is connected to a router, the WAN page shows the correct connections of only the rings that are directly connected to the router. The other source-route bridged token rings in each network are displayed as disconnected segments on the WAN page.

The following figure illustrates this second case.



In all cases, bridge information is not discovered. As a result, discovery treats each interface of a source-route bridge as a separate system on the network. One icon appears in the atlas for each interface of the source-route bridge.

When you have the Traffic Analysis Agent for NetWare installed on one server on each ring of an IPX source-route bridged network, the segment names displayed on the WAN page consist of the IPX network number followed by the MAC address of that server's interface to the ring. If the Traffic Analysis Agent for NetWare is monitoring more than one interface, the address shown for a ring is the MAC address of the interface monitoring that ring.

Transparent Bridges

Discovery cannot completely discover transparent bridges. It consolidates groups of transparently bridged segments running the same network number into a single segment on the maps.

Configuration Changes

Discovery detects most changes in the network topology, such as the addition, reconfiguration, or deletion of interfaces, resulting in changes being made to the atlas. However, if you remove the system from the network, it is not detected unless you move it to another location in the network.

File-Based Discovery

The enhancement to the IPGroper.NLM allows you to use the DiscNodes.txt file to specify the IP Address and mask of a set of nodes to be discovered. The information about the nodes is obtained through SNMP.

The IPGroper NLM must be loaded with specific options that enable it to receive inputs from the DiscNodes.txt file. If these options are not provided, the NLM will discover without taking input from the file. Prior to starting the discovery, the DiscNodes.txt file must be placed in the ZFS-INSTALL-DIR/MWSERVER/NMDISK directory. After the initial discovery, if you want more nodes to be discovered, you must create a new DiscNodes.txt file with the new node entries and place it in the same directory. These nodes will be queried in the next discovery cycle.

The DiscNodes.txt input file has the following format for individual IP addresses:

- ◆ Individual IP Address specification format:

IPAddress <, SubnetMask>

- ◆ Specifying addresses using regular expressions

IPAddress <, SubnetMask>

IPAddress -> AddressPattern

Characters allowed in AddressPattern include the numerals 0-9; the period (.); the question mark (?), which represents one character; and the asterix (*), which represents more than one character, up to a maximum of three.

- ◆ Wildcard characters are not allowed in the subnet mask.

164.99.149.*	All addresses in the range from 164.99.149.1 to 164.99.149.254
164.99.14?.*	all addresses in the range from 164.99.140.1 to 164.99.149.254
164.99.149.?	all addresses in the range from 164.99.149.1 to 164.99.149.9

NOTE: 164.99.149.0 does not come into the range. ? does not stand for 0 if it is the only letter in the octet.
164.99.149.1?0 - all addresses in the range from 164.99.149.100 to 164.99.149.190. Here ? stands from 0

- ◆ In the text file, any line that begins with a " # " is treated as a comment line.

File-based discovery can be used in the following two scenarios:

- ◆ [“Discovering the Nodes Specified in the file” on page 84](#)
- ◆ [“Discovering the Nodes with Other Discovery Modules” on page 85](#)
- ◆ [“Discovering the Nodes with Other Discovery Modules” on page 85](#)

Discovering the Nodes Specified in the file

By default, the Zfs installation loads the NXPCON utility with all the discovery modules running and with file-based discovery enabled.

To discover only the nodes specified in the input file:

- 1** In NXPCON, click Configuration Options > Discovery Modules.
- 2** Select Individual Discovery Modules > press Enter.
- 3** Select No to unload the modules > press Enter.
- 4** Press Esc to exit the Discovery Modules dialog box.
- 5** Click Yes to save changes.
- 6** Click Configuration Options > IP Discovery.
- 7** Select IP Host Discovery > Press Enter.
- 8** Select Enable IP Host Discovery > press Enter.
- 9** Select No to disable autodiscovery of the IP workstation.
- 10** Make sure that the Enable File-Based Discovery option is set to Yes.
- 11** Press Esc to exit the IP Host Discovery dialog box.

- 12** At the Management server prompt, unload NetExplorer by entering **unxp**.
- 13** Reload the NetExplorer modules by entering **netxplore**.

Discovering the Nodes with Other Discovery Modules

By default, the Zfs installation will start all the discovery modules along with the file-based discovery. Use the following procedure to individually select the modules that need to be started or to change the configuration.

To discover only the nodes specified in the input file:

- 1** In NXPCON, click Configuration Options > Discovery Modules.
- 2** Select Individual Discovery Modules > press Enter.
- 3** Select Yes or No to load or unload each module > press Enter.
- 4** Press Esc to exit the Discovery Modules dialog box.
- 5** Click Yes to save changes.
- 6** Click Configuration Options > IP Discovery.
- 7** Select IP Host Discovery > Press Enter.
- 8** Select Enable IP Host Discovery > press Enter.
- 9** Select No to disable Auto Discovery of the IP workstation.
- 10** Make sure that the Enable File-Based Discovery option is set to Yes.
- 11** Press Esc to exit the IP Host Discovery dialog box.
- 12** At the Management server prompt, unload NetExplorer by entering **unxp**.
- 13** Re-load the NetExplorer modules by entering **netxplore**.

Using Command Line Options for IPGROPER

The IPGROPER.NLM has three command line options to discover nodes specified in the DISCNODES.TXT file.

Command Line	Explanation
/Fonly	<p>Specifies that only the nodes specified in the DISCNODES.TXT file must be discovered.</p> <p>You can set this option, when you have set No for the Enable IP Host Discovery option and Yes for the Enable File-Based Discovery option.</p>
/Falso	<p>Specifies that the nodes specified in the DISCNODES.TXT file must be discovered along with the other nodes that IPGROPER will discover.</p> <p>You can set this option, when you have set Yes for both Enable IP Host Discovery option and Enable File-Based Discovery option.</p>
/Flog	<p>Logs all the errors and events that occur during the discovery of the nodes. The errors and the events will be logged in the <zfs_install>\mserver\nmdisk\discnodesbak\discnodeslog.log file. You must manually enter this command line option in the NETXPLORE.NCF file.</p> <p>IMPORTANT: The DISCNODESLOG.LOG file will be created only if you have specified the /fonly or the /flog</p>

Effects of Discovery on Maps

The Atlas Manager on the management server creates an atlas database as the topology database is populated and the information is displayed as maps on ConsoleOne. The WAN page displays all the Area pages and the connecting routers between them. The Area pages display the segments and the connecting routers.

The discovered systems are placed on the Area pages of the atlas based on the connecting routers or bridges. The Islands page contains segments for which routers have not yet been discovered. The Atlas Manager relocates the segments to the correct pages when connecting routers are discovered.

Review the following sections for more information on the effects of discovery on maps:

- ◆ “Name Source Priority” on page 86
- ◆ “Representation of Systems in the Atlas” on page 86

Name Source Priority

As discovery cycles proceed and more information is discovered, the names displayed in the maps can change. Different priorities are given to names, depending on the source of the name information.

To determine how to display the name of the discovered object, the Atlas Manager uses the following list in the order shown:

1. User Defined Name
2. DNS Name
3. eDirectory Name
4. Bindery Name
5. SNMP Name
6. IP Address
7. IPX Address
8. MAC Address

Representation of Systems in the Atlas

When representing a system in a map, the Atlas Manager refers to the following list of services in the order shown. As soon as it associates the first service with the node, it displays it without looking for further matches. The icon may change if a service with a higher priority is detected later during discovery.

Priority Number	Icon	Description
1.		NetWare server running the server management agent software
2.		Windows NT server running the server management agent software

Priority Number	Icon	Description
3.		SFT III server running the MS engine
4.		Server running file server software
5.		Router running IP service
6.		Router running IPX service
7.		A switch or a bridge
8.		Server running the discovery process
9.		Server running the topology database
10.		NetWare or Windows NT server running the traffic analysis (LANalyzer) agent software
11.		Server running Remote Monitoring
12.		Server running Remote Monitoring II
13.		Server running print server software
14.		Server running IP software
15.		Server running NetWare Connect software

Priority Number	Icon	Description
16.		Router
17.		Printer
18.		IP workstation
19.		IPX workstation
20.		Others

If a system has either an IPX or IP router service, the Atlas Manager considers it a router and displays it on the appropriate pages and segments.

Setting Up Discovery

The discovery software on a management server automatically discovers the nodes on your network. Network nodes include servers, desktops, routers, hubs, switches, and any other network devices. The Consolidator on the server populates the database with the discovered data. The Atlas Manager on the server reads the database and creates the atlas.

ZfS allows discovery in two different environments:

- ◆ Pure IP environment
- ◆ IP/IPX environment

You must have IP enabled between ConsoleOne and the management server.

Before starting discovery, you must verify the following configurations to ensure that the discovery system is complete:

- ◆ Ensure that the router to which ZfS Server is attached is specified as the seed router in NXPCON. If necessary specify, additional IP routers also. For more information on specifying seed router and additional IP routers, see [“Specifying a Seed Router and Additional IP Routers” on page 96](#)
- ◆ Ensure that the community strings used for all the devices to be managed are specified in NXPCON. For more information on changing SNMP community strings, see [“Changing the SNMP Community String” on page 93](#).
- ◆ Ensure that the ZfS Server is privileged to query the routers in your network if the routers are configured to restrict access to only specified IP addresses. For more information on IP router discovery, see [“IP router discovery on IP networks only.” on page 69](#).

- ◆ If you want to restrict the scope of IP or IPX discovery, specify proper scoping entries. For more information on changing the discovery scope, see [“Changing the Discovery Scope” on page 93](#).
- ◆ Ensure that the DNS configuration file `SYS:\ETC\RESOLV.CFG` has a valid DNS server's IP address. If a valid DNS server is not specified, discovery will fail to discover the DNS names of hosts.
- ◆ For effective discovery, ensure that the Traffic Analysis Agent is installed and running on each network segment that you want to discover and manage. Also, ensure that the names and addresses of these agents are specified in `NXPCON`. For more information on specifying Traffic Analysis Agents, see [“Specifying Traffic Analysis Agents to Be Queried by NXPLANZ” on page 96](#).
- ◆ If a MAC address is being associated with different network numbers, all such network numbers will be merged into a single segment. To avoid the merger, you must specify all such MAC addresses in upper case in the `installation_directory\MMS\MWSERVER\BIN\CONSOLIDATOR.INI` file.

In `CONSOLIDATOR.INI`, specify the MAC address as a key value pair in the `[DuplicateMacAddress]` section.

A sample `CONSOLIDATOR.INI` is as follows:

```
[DuplicateMacAddress]
mac1="00C04F59910D"
mac2="00C04F5991AB"
...
key_name=value
```

In `CONSOLIDATOR.INI`, ensure that the keys are unique.

Before starting the MMS server, edit the `ZENWorks\MMS\MWSERVER\PROPERTIES\SLOADER.PROPERTIES` file to append the `ARGUMENTS` value under `TOPOLOGY MANAGER` with the following entry: `-ini "installation_directory\MMS\MWSERVER\BIN\CONSOLIDATOR.INI"`.

The following tasks will start discovery initially and help you customize discovery to meet your organization's needs:

- ◆ [“Starting Discovery” on page 90](#)
- ◆ [“Checking the Status of Initial Discovery” on page 90](#)
- ◆ [“Checking the Results of Discovery” on page 91](#)
- ◆ [“Changing the Default Configuration” on page 92](#)
- ◆ [“Configuring the Java Processes” on page 97](#)
- ◆ [“Unloading the Management Server” on page 98](#)

Starting Discovery

Discovery starts automatically when the discovery software is loaded on the management server.

To manually start autodiscovery and load the back-end services (management site services), refer to the steps in [“Installing and Setting Up Management and Monitoring Services”](#) in the *Installation* guide.

Restarting the Management Server

If you bring down the management server (for example, for maintenance), the restart affects discovery in the following ways:

- ◆ Each time you reload the discovery modules, a new version of NETXPLOER.DAT is created.
- ◆ The initial discovery cycle starts again.
- ◆ The Consolidator processes all the discovery data again as ZfS rediscovers the network.

To unload the discovery modules:

- 1 At the NetExplorer server, enter **unxp**.

To load the discovery modules:

- 1 At the NetExplorer server, enter **netxp**.

Checking the Status of Initial Discovery

As discovery progresses, your topology maps in ConsoleOne reflect the discovered data. However, in a large network, it might take a day or two before the initial discovery is complete.

The easiest way to determine whether initial discovery is complete is to use the NXPCON utility on the management server and check the status of each NetExplorer module. Each module must complete at least one full cycle to draw a complete map.

To view the discovery status, look at the discovery status fields at the top of the NXPCON screen. See [“Using the Discovery Configuration Utility” on page 92](#) for information about how to access this screen.

The NXPCON main screen gives you the information you can use to monitor the status of discovery.

The following information is displayed:

- ◆ **NetExplorer Up Time:** Shows the time since NetExplorer started running.
- ◆ **NetExplorer System Status:** Shows the overall status. It can have one of the following values:
 - ◆ Waiting to start - Waiting for one or more of the discovery modules to start.
 - ◆ Running - Discovery modules are running.
- ◆ **Module Status:** Shows the status of each module and the number of cycles each module has completed. The module status can be one of the following values:
 - ◆ Not Loaded - Module is not loaded.
 - ◆ Waiting to Start - Module is loaded but not started.
 - ◆ Running - Module is running and collecting data.
 - ◆ Suspended - Module is suspended because it reached the end of the schedule in which it was running.
 - ◆ Completed - Module completed a discovery cycle.
 - ◆ Unknown - NetExplorer cannot obtain the module status. (This is usually seen if the module is not loaded.)

Checking the Results of Discovery

When the Consolidator has finished updating the database after the initial discovery, verify if the network topology is accurately represented on the maps.

NetExplorer might not have discovered the type if a node is not on the map. If a node does not appear in the correct segment, NetExplorer may not have received sufficient information to place it correctly. For more information, see [“What Is Discovered” on page 75](#). The following characteristics are captured:

- ◆ IP - Discovers IP routers; IP hosts; IP services such as HTTP, Telnet, SMTP, DNS, FTP; and DHCP.
- ◆ IPX - Discovers IPX workstations, IPX routers, and IPX services (file, print, any other Service Advertising Protocol [SAP]).
- ◆ Subnet mask
- ◆ Services
- ◆ eDirectory names and tree
- ◆ DNS Names

The Consolidator on the management server communicates with NetExplorer to obtain network discovery data. The Consolidator reads the NETEXPLOR.DAT file and populates the database.

IMPORTANT: The NETEXPLOR.DAT file is reset every time you restart NetExplorer.

The Consolidator communicates with two Java* components: the Bridge Agent and the SN3 agent. The Bridge agent retrieves bridges present in the network and the related topology of the network. The SN3 agent does SLP-based discovery for NetWare 5.x servers and gets the corresponding eDirectory name for each IP and IPX address discovered.

IMPORTANT: NetExplorer and the Consolidator can run independent of each other on the management server.

NetWare 5.x servers are discovered faster because NetWare 5.x supports the Service Location Protocol (SLP).

Ensuring Complete Discovery

IPX workstations are discovered with a username if the user is logged in to or attached to a NetWare server running management agent software. To ensure that the usernames for IPX devices and workstations on your network can be discovered, install a management agent on all NetWare servers where users log in.

If you want NetExplorer to discover AppleTalk* devices, you need to install the NetWare LANalyzer Agent on one server on each segment.

Changing the Default Configuration

The discovery software is installed with default configuration designed to work in most environments. However, if your network or the data on your database is not discovered, you need to reconfigure discovery.

Read the following sections for more information:

- ◆ [“Using the Discovery Configuration Utility” on page 92](#)
- ◆ [“Choosing Which Discovery Modules to Load” on page 92](#)

- ◆ “Changing the SNMP Community String” on page 93
- ◆ “Changing the Discovery Scope” on page 93
- ◆ “Specifying Traffic Analysis Agents to Be Queried by NXPLANZ” on page 96
- ◆ “Specifying a Seed Router and Additional IP Routers” on page 96

Using the Discovery Configuration Utility

You can use the NXPCON utility on the management server to change the discovery configuration. For example, you can change the scope of discovery or view the status of the initial discovery process.

To access the NXPCON utility:

- 1** Access the server console on the management server either directly from the server prompt or remotely.
- 2** If the discovery modules are already loaded on the server, click the NetExplorer Console Utility option in the Available Screens window.

or

If the discovery modules are not loaded, enter **netexplor** at the server prompt.

NXPCON is loaded automatically when NetExplorer is loaded and is accessible at the management server.

If NXPCON is not loaded on your management server, check to see if NetExplorer is running. If NetExplorer is running, enter **load nxpcon** at the system console prompt. If NetExplorer is not running, enter **netexplor** at the system console prompt.

Choosing Which Discovery Modules to Load

By default, the ZfS installation loads the NXPCON utility with all modules running. If you are not using IPX on your network, you can configure NXPCON to not load the NXIPX module.

IMPORTANT: Make sure TCP/IP is bound to at least one of your server’s network boards.

To view or modify which modules are being loaded:

- 1** In NXPCON, click Configuration Options > NetExplorer Modules.
- 2** Select the field you want to change > press Enter.
- 3** Select Yes or No to load or unload the module > press Enter.
- 4** Press Esc to exit the NetExplorer Modules dialog box.
- 5** Click Yes.

You can enable IP host discovery or file-based discovery. To enable or disable:

- 5a** Select Configuration Options > IP Discovery.
- 5b** Select IP Host Discovery or File Based Discovery > Press Enter > Press Yes to enable or press No to disable the discovery option.
- 6** At the management server prompt, unload NetExplorer by entering **unxp**.
- 7** Reload the NetExplorer modules by entering **netexplor**.

Changing the SNMP Community String

In ZFS, the default community string is PUBLIC. If your organization's SNMP community string is not PUBLIC, reconfigure the SNMP community string in NXPCON.

NOTE: In order to prevent burdening the routers, some organizations add one more level of control by allowing only certain IP addresses to do SNMP queries to the routers. If this is true in your organization, make sure that the IP address given to the ZFS sever is privileged to query the routers in the network. Otherwise, the discovery will not be complete and incomplete network information will appear under "Islands" in the atlas.

To view, add, modify, or delete SNMP configuration information, such as community strings used for IP and IPX discovery:

- 1** In NXPCON, click Configuration Options > SNMP.
- 2** In the SNMP dialog box, click Edit Community Name List.
- 3** To add a community string, press Insert.
or
To modify a community string, click the community string > press Enter.
or
To delete a community string, click the community string > press Delete.
- 4** Press Esc > click Activate Changes from the Configuration Options window.
- 5** Respond to the prompts accordingly.

For information about other configuration options in the SNMP window, see [“Using the Discovery Configuration Utility” on page 92](#), or ConsoleOne online help.

Changing the Discovery Scope

By default, NXPCON is set to discover all IPX and IP networks. You can, however, limit the discovery scope.

You could, for example, limit discovery to discover the IPX addresses or the IP subnet addresses. If you are managing a large network, by setting the scope of discovery, you will be limiting the discovery to a section of your network, which will reduce the network traffic and in turn make your atlas more manageable

If you do not accurately specify the scope of discovery, you will not be able to discover your target device. Therefore it is imperative to specify in the scope, all the devices that are present in the path leading to the target device you want to discover.

For example, consider the following scenario:

Your discovery server D1 is connected to network N1. Router R1 connects network N2 with N1. Assume you need to discover network N2. To do this, the following entries need to be set in the scope:

- ◆ Discovery server D1 with subnet mask 255.255.255.255
- ◆ Router R1 with subnet mask 255.255.255.255
- ◆ Network N2 with its appropriate subnet mask number.

In this scenario, network N2 can be reached from the discovery server through Router R1, and therefore R1 needs to be in the scope even if the user is not interested in the network N1 that R1 is routing.

After initial discovery, until you reset the database, nodes remain in the database even if they have been removed from the network.

Changing the discovery scope does not affect devices that are already in the database due to prior runs of discovery. In particular, devices that were discovered due to a wider scope (or no scope) will not be removed when a restrictive scope is set for later runs of discovery. If it is desired that the atlas shows only those devices that fall in scope, the database needs to be reset to ensure that segments and devices that are out of scope do not appear in atlas. Note that the database being reset would result in loss of data like alarms and alarm disposition unless they are migrated. Alternatively, if the number of such devices which are out of scope is very small, the user can manually delete them from the database using the Database Object Editor.

You can restrict the scope of IP or IPX discovery by entering the IPX network numbers or IP address ranges specified by the mask fields you want to discover. To view or restrict the IP or IPX scope:

- 1 In NXPCON, from the Configuration Options window, click Discovery Scope.
- 2 Select IP Discovery Scope or IPX Discovery Scope.
- 3 Press Enter to view or configure the scope of your discovery.
- 4 Press Insert to add a new IP or IPX discovery scope entry.

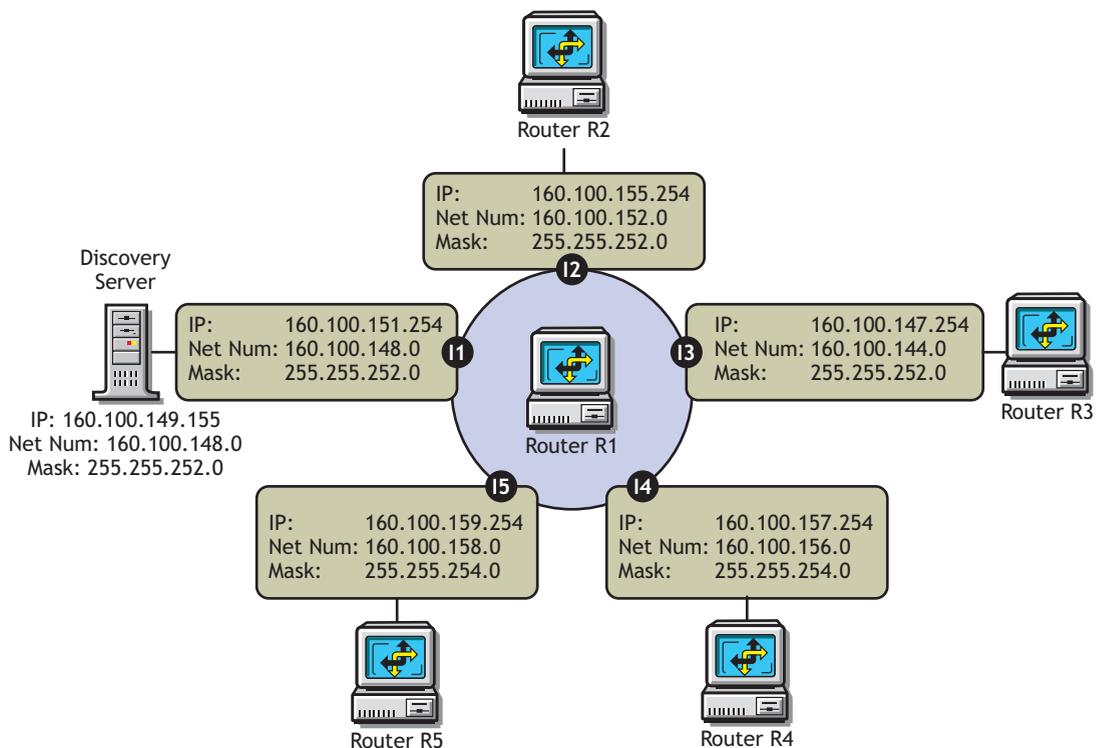
or

Press Enter to modify a discovery scope entry.

or

Press Delete to delete a discovery scope entry.

For IP Networks: Discovery scope is tightly bound to the network numbers. The scope can be restricted by specific networks as illustrated in the following diagram.



Case 1: To exclude 160.100.148.0 and discover the other four networks, specify the scope as:

160.100.149.155, 255.255.255.255

160.100.151.254, 255.255.255.255

160.100.144.0, 255.255.252.0

160.100.152.0, 255.255.252.0

160.100.156.0, 255.255.254.0

160.100.158.0, 255.255.254.0

The 255.255.255.255 mask for the ZfS server and the router interface on the local network acts as a machine specific scope. This prevents other machines in the network 160.100.148.0 from being discovered.

Case 2: To discover only the local network 160.100.148.0, specify the scope as:

160.100.148.0, 255.255.252.0

The network 160.100.148.0 (Mask: 255.255.252.0) has IP addresses in the range 160.100.148.1 to 160.100.151.254.

Consider a case where all the important servers in your network have IP addresses in the range 160.100.149.1 to 160.100.149.254. You might specify the following scope:

160.100.149.0,255.255.255.0

The above scope is not allowed by the discovery system. You cannot set a scope to discover only a part of the subnet. You will have to set the entire subnet in scope.

Case 3: To discover only 160.100.156.0 and 160.100.158.0 scope should be given as:

160.100.149.155, 255.255.255.255

160.100.151.254, 255.255.255.255

160.100.156.0, 255.255.254.0

160.100.158.0, 255.255.254.0

Replacing the last two scoping entries with a single entry 160.100.156.0, 255.255.252.0 might not have the same effect.

You cannot create a single scoping entry to cover two or more subnets. You have to create a scope for each subnet.

For IPX Networks: Restrict the scope to the IPX networks to be discovered by entering a single IPX network number and a mask.

The mask indicates which part of the network number needs to match. An F in the mask means that the corresponding digit must match; a 0 (zero) means that no match is required.

For example, network number 12340000 and mask FFFF0000 will match any network number starting with 1234.

Network number C00000FF and mask FF0000FF will match any network number starting with C0 and ending with FF, such as C01234FF or C00000FF.

- 5** Enter the address and mask for your discovery scope.
- 6** Press Esc > click Yes to save changes to the configuration file.
- 7** Press Esc to return to the Discovery Scope window.

- 8 Unload and reload the NetExplorer modules or restart your management server for the changes to take effect.

Specifying Traffic Analysis Agents to Be Queried by NXPLANZ

Traffic analysis agents in your network are usually discovered by the NXPLANZ module. If SLP is disabled or if SAP packets are filtered by the routers in your network, NXPLANZ might not be able to discover all the Traffic Analysis Agents in the network.

To specify Traffic Analysis Agents to be queried by the NXPLANZ module:

- 1 In NXPCON, click Configuration Options > NXPLANZ Discovery.
- 2 To add an agent, press Insert.
- 3 Enter the address and mask for your discovery scope.
- 4 Press Esc > click Yes to save changes to the configuration file.
- 5 Unload and reload the NetExplorer modules or restart your management server.
- 6 To modify an agent, select the agent > press Enter. Modify the required information.
- 7 To delete an agent, select the agent > press Delete.

Specifying a Seed Router and Additional IP Routers

Seed router is the router to which ZfS Server is connected. For router discovery to be effective, always specify the seed router using NXPCON and ensure that ZfS server can query the seed router by specifying the proper community name in NXPCON.

You need to specify additional IP routers if you want to discover one part of your network and the ZfS server does not have access to one of the intermediate routers.

To specify a seed router or additional IP Routers:

- 1 In NXPCON, click Configuration Options > IP Discovery > IP Router Discovery.
The default for IP Seed Router is *<local>*, which is the ZfS server.
- 2 To add a seed router, select IP Seed Router and press Enter.
- 3 Enter the IP address.
- 4 To add additional routers, select Additional IP Routers and press Enter.
- 5 Enter the IP address.
- 6 Press Esc > click Yes to save changes to the configuration file.
- 7 Unload and reload the NetExplorer modules or restart your management server.

Configuring the Java Processes

The following are the three Java processes of the discovery system:

- ♦ Topology Manager
- ♦ Bridge Discovery
- ♦ SN3 Discovery

These Java processes form a part of the ZfS site server and exist as sections in the SLOADER.PROPERTIES file in the *install_path\ZENWORKS\MMS\MWSERVER\PROPERTIES* directory. They are specified in the following format:

```
[Topology Manager]
Name = Topology Manager
Load Option = auto
Other options
```

To configure the Java processes:

1. Change the value of the Load Option from Auto to Manual to prevent the process from starting the next time you type the SLOADER command on the server.

IMPORTANT: If you modify the SLOADER.PROPERTIES file after you start the ZfS Site server, you must restart the ZfS Site server for the changes to take effect.

2. Do not change the Load Properties and the Load Sequence options in the SLOADER.PROPERTIES file. These options are necessary for the ZfS site server to work correctly.

Customizing Starting and Stopping Discovery

You can choose to stop or start the discovery NLM files or the Java discovery processes without affecting the other services of the site server, such as the Alarm Manager Service.

Stopping and Starting the Discovery NLM Files

To stop the discovery NLM files, enter **UNXP** at the server console.

To start all the discovery NLM files, enter **NETXPLO**R at the server console.

NOTE: You cannot start the Discovery NLM files if the Java processes are running. Stop the Java processes and then type **NETXPLO**R at the server console to start all the discovery NLM files.

Stopping and Starting the Java Discovery Services

To stop the discovery NLM files, enter **STOPDIS** at the server console.

To start all the discovery NLM files, enter **STARTDIS** at the server console.

You can customize starting or stopping any of the Java discovery processes at any point in time. For example, you decided not to run the Bridge discovery initially but decide to run it anyway. In such a scenario, you need not stop all the services and restart them. You can edit the STARTDIS.NCF file in the *\ZENWORKS\MMS\MWSERVER\BIN* directory, which has the following contents:

```
MWSETENV.NCF
```

```
java -Xbootclasspath/p:$mwxbpath -classpath
$MMSCP;$CLASSPATHcom.novell.utility.servicemanager.ui.Start "Topology
Manager" "Bridge Discovery" "SN3 Discovery" <ip address of the server> sloader
```

In the above file, the Java discovery process names like SN3 Discovery must match the names of the sections in the SLOADER.PROPERTIES file. By changing just the names in the NCF files, you can create similar NCF files to selectively stop and start the Java discovery services. For example, if you want to start just the Bridge discovery process:

1. Create a STARTBRI.NCF file with the following contents:

```
MWSETENV.NCF
```

```
java -Xbootclasspath/p:$mwxbpath -classpath  
$MMSCP;$CLASSPATHcom.novell.utility.servicemanager.ui.Start "Bridge  
Discovery" <ip address of the server> sloader
```

2. Copy the STARTBRI.NCF file to the \ZENWORKS\MMS\MWSERVER\BIN directory.
3. Run the STARTBRI.NCF file to start the Java discovery bridge service.

For example, to stop the Java discovery process for the SN3 Agent:

1. Create a STOPSN3.NCF file with the following contents:

```
MWSETENV.NCF  
  
java -Xbootclasspath/p:$mwxbpath -classpath  
$MMSCP;$CLASSPATHcom.novell.utility.servicemanager.ui.Stop "SN3  
Discovery" <ip address of the server> sloader.
```

Unloading the Management Server

To unload the management server:

- 1 If restarting the server is not feasible, make sure all ServiceLoader processes are exited. At the server console prompt, enter

```
stopService.ncf
```

```
java -show
```

TIP: You may sometimes see a process displaying the status "exiting" in the java -show display. If this condition persists, restart the server.

You can use **java -exit** if you can terminate all other Java processes. Unload Java, if all the services are not closed.

- 2 Enter **unmw** to unload all ZfS components.
- 3 Switch to the Sybase* process by pressing Ctrl+Esc > enter **q** to terminate the Sybase database engine.

Managing the Atlas

After the initial discovery, you can stop discovery running on the management server. You can, however, continue to access the database through the Atlas Manager. The discovery cycle starts again the next time NetExplorer is up. The Consolidator populates the database and the Atlas Manager automatically updates the atlas pages.

Depending on the size of your network, writing data from the initial discovery cycle can take few minutes to several days. Subsequent discovery updates to the database require substantially less time.

- ♦ [“Using the Atlas” on page 99](#)
- ♦ [“Using Unified Views” on page 101](#)

Using the Atlas

When ZfS is first installed, the server module of the Atlas Manager is automatically installed on the management server, and the client module of the Atlas Manager is installed on ConsoleOne.

The Atlas Manager on the management server creates a system atlas and provides a graphical view of the database at the console.

The Atlas Manager on the server reads the database and provides two different views of the database at ConsoleOne: the Console view and Atlas view. Both views provide information about the discovered network topology, the physical location of nodes, node configuration information, and alarm information.

The following sections gives you an understanding about using the atlas:

- ◆ “Accessing the Atlas” on page 99
- ◆ “Assigning Roles to Help You Manage the Atlas” on page 100
- ◆ “Using the Atlas to Troubleshoot” on page 100

Accessing the Atlas

You can access the ZfS atlas from ConsoleOne. Open ConsoleOne and double-click the ZfS Domains namespace, then expand the domain. The system atlas appears.

The following table describes a ZfS atlas consisting of three different pages:

Atlas Pages	Icon	Description
WAN page		Summarizes the entire network, illustrating the WAN-related network topology. Your atlas, typically, has a single WAN page.
Area page		Displays segments on your network. An atlas can have several area pages. For example, areas can be divided based on the geographic location of the network. If a company in San Jose has an overseas branch in Germany, you can divide your network into Area1 for the San Jose network and Area2 for the Germany network.
Islands page		Consists of segments with an undetermined connectivity. During discovery, the Islands page is a placeholder for network objects that are not completely discovered. An atlas has a single Islands page.

Customizing Your Atlas View

You can customize your atlas view in four different ways:

- ◆ Insert a custom bitmap as the background on an atlas page.
- ◆ Change the position of a node on an atlas page by dragging it.
- ◆ Display objects by an alternate name.

Assigning Roles to Help You Manage the Atlas

ZfS lets you assign roles to manage the atlas. By assigning roles, you can restrict the user from performing specific operations on that object.

TIP: The atlas displays maps based on your role on the network. For example, if your role is restricted to managing certain servers in segment A and B, your atlas will contain only those servers in segments A and B.

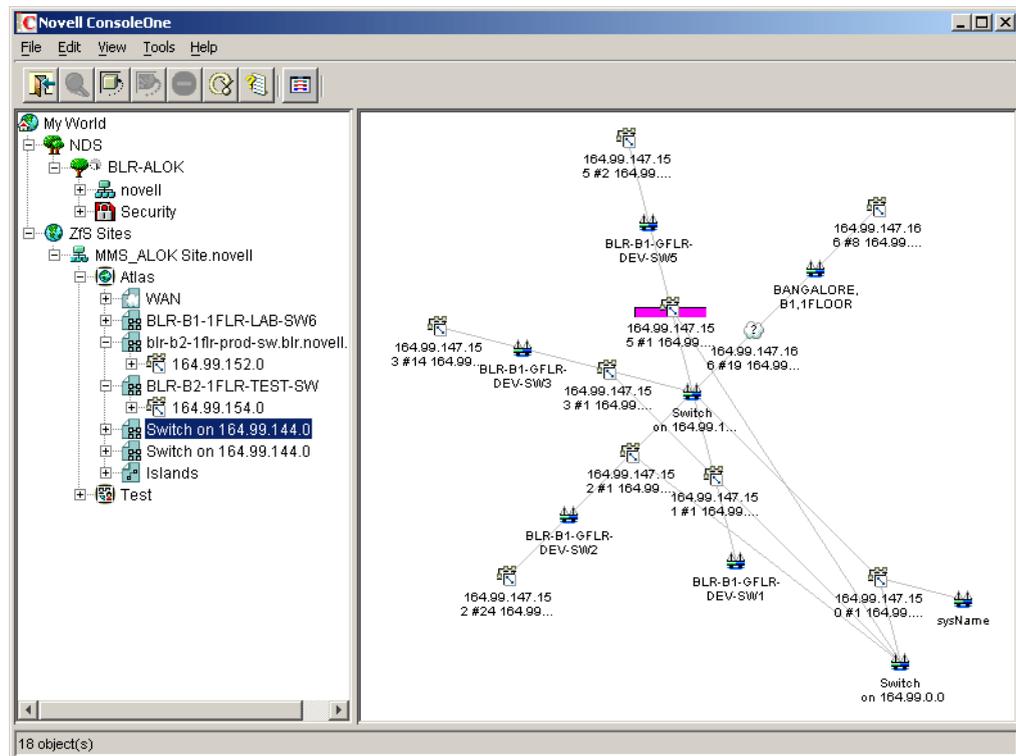
You can perform the following tasks on any atlas page (WAN, Islands, or Area page) when the Atlas view is displayed on ConsoleOne.

Tasks	Comments
Open	Opens the page
Import	Inserts a custom wallpaper
Save	Updates the changes in the database
Print	Prints the page
Rename	Renames the page
Layout	Displays the page with a different focal point

Using the Atlas to Troubleshoot

By setting the alarm disposition to save alarms in the database, ZfS maps can alert you to alarm conditions on the network. Alarms are of type severe, major, or minor alarm on a segment or node. Upon recognizing any of these alarms, the ZfS ConsoleOne displays a bell-shaped alarm icon above the object. The alarm status is propagated up the hierarchy. For example, if a server has an alarm of type severe, the segment and the page containing the server will display the corresponding alarm icon. For information about alarms, [“Managing the Alarm Management System” on page 109](#).

The following figure shows the atlas namespace in ConsoleOne:



Using Unified Views

The Unified view service is a service that acts as a filter on the atlas. Using the Unified view, you can filter for a list of devices or segments of a particular type. The Unified view allows easy navigation and quick operations to check the highest severity of the alarms present on a particular node or segment.

The following are the two types of Unified view provided:

- ◆ “Unified View for Devices” on page 101
- ◆ “Unified View for Segments” on page 102

Unified View for Devices

You can view All, Manageable, or Unmanageable devices in this view. For a corresponding device type, a device is said to be manageable if the list of MIBs implemented by the device satisfies the Manageability_definition property in the UnifiedView.ini file in the ZENWORKS\MMS\MWSERVER\BIN directory. The Manageability_definition property can be updated with a valid boolean expression of MIB names.

Following are the device types that you can filter:

- ◆ All (all types of devices)
- ◆ Netware Servers
- ◆ NCP Print Servers
- ◆ TCP Services
- ◆ Printers
- ◆ IP Routers
- ◆ Switches/Bridges
- ◆ IPX Routers
- ◆ Windows NT Servers

To filter the devices:

- 1** In the atlas, select View > Unified View for Devices.
- 2** From the first drop-down list, select All to list all the devices
or
Select Manageable to list the manageable devices
or
Select Unmanageable to list the unmanageable devices.
- 3** From the second drop-down list, select a device type.
- 4** Click Show.

The Unified view will display the list of the devices. The tabular column in the Unified view contains the following information.

- ◆ The icons associated with the devices.
- ◆ The MIBs implemented by the device. If the device does not implement any MIBs the column will specify "No MIBs implemented" for that device.

- ◆ The maximum severity of the alarms against the devices. To view the legend for the alarm, select the alarm legend button on the toolbar.

Unified View for Segments

You can view All, Manageable, or Unmanageable segments in this view. For a corresponding segment type, a segment is said to be manageable if the list of MIBs implemented by at least one device in that segment satisfies the Manageability_definition property in the UnifiedView.ini file. The Manageability_definition property can be updated with a valid boolean expression of MIB names. The following are the segment types you can set filter for:

- ◆ All (all types of segments)
- ◆ Ethernet
- ◆ Frame Relay
- ◆ IPX Compatibility Mode
- ◆ Token Ring
- ◆ X.25
- ◆ PPP
- ◆ ATM
- ◆ FDDI

To filter the segments:

- 1** At the Atlas level, select View > Unified View for Segments.
- 2** From the first drop-down list, select All to list all the segments
or
Select Manageable to list the manageable segments
or
Select Unmanageable to list the unmanageable segments.
- 3** From the second drop-down list, select a segment type.
- 4** Click Show.

The Unified view will display the list of the segments. The tabular column in the Unified view contains the following information.

- ◆ The icons associated with the segments.
- ◆ The name of the segment.
- ◆ The maximum severity of the alarms against the segments. To view the legend for the alarm, select the alarm legend button on the toolbar.

4

Understanding Alarm Management

The ZENworks[®] for Servers (ZfS) Alarm Management System (AMS) alerts you to important events like the SNMP traps, threshold alarms, network discovery events, and ping and connectivity testing faults occurring on your network. This lets you proactively resolve network problems and receive updates on events occurring on your network.

Alarm icons are anchored to objects displayed in ConsoleOne[®]. The icons change color to depict the level of severity, notifying you of potential problems. The events are reported in the Active Alarm view, and each event is categorized and displayed with a corresponding alarm icon.

AMS will process any device on the network that supports SNMP-standard trap notification. For example, for all NetWare[®] servers on which the Management Agent for NetWare is installed, notifications of server breakdowns, overloads, and configuration changes are sent to the management server for processing and then made available for viewing at a ZfS ConsoleOne.

You can enable and disable alarms and set alarm thresholds on baseline statistics for segments and servers (for example, segment alarms for utilization and the total number of packets per second), so that an alarm is generated when the threshold for a statistic is reached. You can also set actions to be performed when an alarm or an event occurs. The actions assigned to an alarm or event are specified in the alarm disposition.

This section contains the following topics:

- ◆ [“Understanding the Alarm Management System” on page 105](#)
- ◆ [“Managing the Alarm Management System” on page 109](#)
- ◆ [“Maintaining the Alarm Management System” on page 123](#)
- ◆ [“Troubleshooting the Alarm Management System” on page 123](#)

Understanding the Alarm Management System

AMS alerts you to network conditions and events. AMS provides you with tools and back-end services to use, distribute, and manage this information. The AMS component is also fully integrated with other ZfS components. It provides access control through the role-based services (RBS) component and report generation through the reporting functions. AMS provides a centralized location for processing and viewing the events and alarms generated by devices and systems throughout your network.

You can view tabular lists of statistical data for active and historical alarms received by AMS from ConsoleOne. This makes it easy to handle alarms and track network events and recurring alarm conditions. In addition, real-time notification of alarms occurring on your network is provided by the following

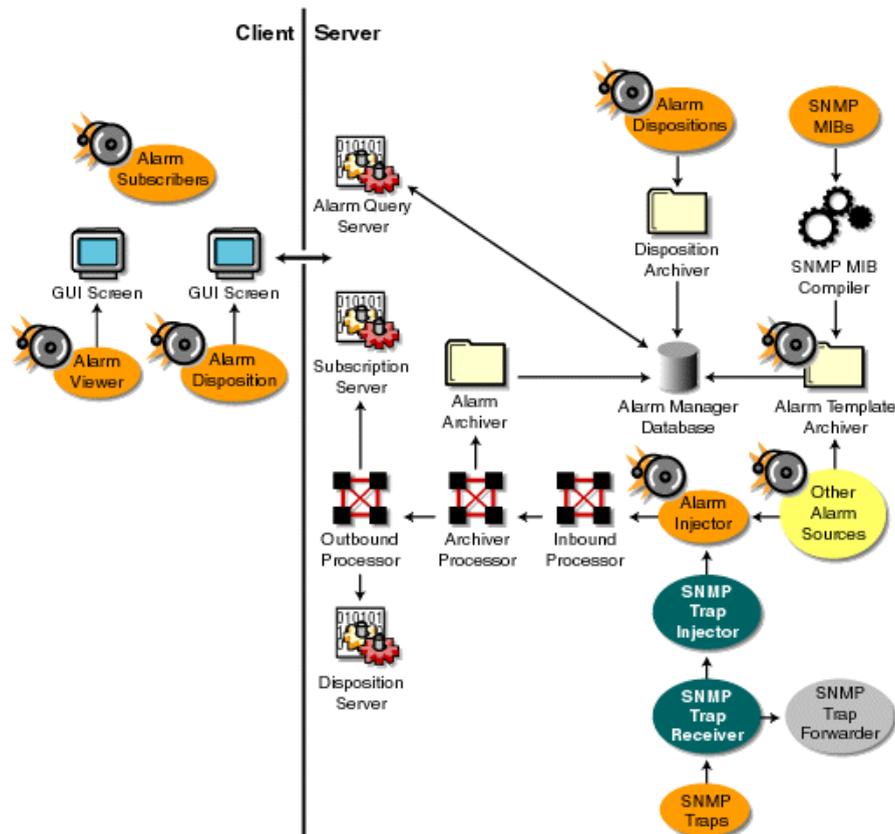
- ◆ Severity level, as displayed by the changing color of the alarm indicators
- ◆ Audible notification
- ◆ Status bar ticker-tape messages

You can also assign an action, such as automatically launching a program when an alarm is received, or sending an e-mail message to notify remote users of events.

Alarm Management System Components

AMS comprised of multiple components for processing, storing, and viewing alarms. All alarms received by AMS are processed and sent to applications that subscribe to them. The ZFS ConsoleOne, by default, subscribes to AMS and receives updates when an alarm is processed. Hierarchical Status Notification (HSN) also subscribes to AMS and changes the color of the atlas map icon accordingly.

The following figure illustrates the AMS components:



The main components that make up AMS are as follows:

- ◆ “SNMP Trap Receiver” on page 107
- ◆ “SNMP Trap Forwarder” on page 107
- ◆ “SNMP Trap Injector” on page 107
- ◆ “Alarm Injector” on page 107
- ◆ “Alarm Processors” on page 107
- ◆ “Alarm Manager Database” on page 107
- ◆ “Archivers” on page 108
- ◆ “Alarm Viewers” on page 108

SNMP Trap Receiver

SNMP receives traps from network management agents and passes them to the SNMP trap forwarder and the SNMP trap injector.

SNMP Trap Forwarder

After the SNMP trap forwarder receives a trap, it checks the alarm manager database to determine whether the trap has an SNMP trap forwarding disposition. If the trap has a forwarding disposition, the SNMP trap forwarder forwards the trap. Otherwise, the SNMP trap forwarder ignores the trap.

SNMP Trap Injector

The SNMP trap injector converts the trap to an alarm and passes the alarm to the alarm injector.

Alarm Injector

The alarm injector receives alarms from the SNMP trap injector and other applications and passes them to the inbound processor.

Alarm Processors

The alarm processors includes processes for receiving, archiving, and dispatching alarms to subscribers. The inbound processor applies alarm templates to incoming alarms. After inbound processing is completed, the alarm is sent to the archive processor, which facilitates logging and storing of the processed alarm data in the alarm manager database. The archive processor sends the alarm to the outbound processor, which in turn sends the alarms to the subscription server and disposition server.

Alarm Manager Database

The alarm manager database, a repository for alarm information, includes the following:

- ◆ [“Processed Alarms” on page 107](#)
- ◆ [“Alarm Templates” on page 107](#)
- ◆ [“Alarm Dispositions” on page 108](#)

Processed Alarms

The processed alarm data that is stored in the alarm manager database is supplied to ConsoleOne through the alarm query server. The alarm data is used for alarm and alarm summary presentation and reporting.

Alarm Templates

Templates are applied to each alarm received by the inbound processor. The alarm template is based on SNMP trap definitions in the MIB or other proprietary definitions for handling the AMS management and display criteria. When you compile an MIB, the trap definitions are used to create an alarm template that provides a method for presenting and managing alarm data. Proprietary

alarms templates are based on proprietary definitions. For example, when a user tries to log in to a server with an incorrect password, an alarm is generated and forwarded to the management server. The management server processes the alarm by identifying the trap object identifier (OID) and assigns the associated alarm template.

A default template is assigned to an SNMP trap sent by a device that does not have a recognizable OID and is categorized as unknown. In order for a trap OID to be recognized by AMS, you need to compile the MIB of the device into the MIB Pool on the management server. For more information, see [“Installing and Setting Up Management and Monitoring Services”](#) in the *Installation* guide.

Alarm Dispositions

Alarm dispositions govern the handling characteristics for each type of SNMP trap or proprietary alarm. AMS allows you to configure the automatic handling of an alarm by defining it in the alarm disposition. The automatic handling functions include specifying an application to launch when an alarm is received, sending e-mail notification, forwarding processed alarms to other ZIS management servers, and forwarding SNMP traps to other network management systems. You can also set options for alarms, such as audible beeps.

Archivers

The following three archivers add data to the alarm manager database:

- ◆ [“Alarm Archiver” on page 108](#)
- ◆ [“Disposition Archiver” on page 108](#)
- ◆ [“Template Archiver” on page 108](#)

Alarm Archiver

The alarm archiver stores alarm statistics and data in the alarm database. By default, all alarms are archived. If you do not want an alarm archived, you can edit the alarm disposition to disable archiving of the alarm. See [“Archiving Alarm Statistics” on page 122](#) for more information.

Disposition Archiver

The disposition archiver receives the alarm disposition from the Disposition Console and saves it in the alarm manager database.

Template Archiver

The template archiver receives alarm templates from a MIB compiler and saves them in the alarm manager database.

Alarm Viewers

ConsoleOne displays three views of alarm data: the Active Alarm view and the Historical Alarm view and the Alarm Summary view.

The Active Alarm view displays statistics in ConsoleOne for events occurring on your network. Alarms displayed in the Active Alarm view can either be owned by you or assigned to a group. The tasks that you can perform on an alarm from this view depend on the access rights allowed through the role-based services. The Active Alarm view continually appends incoming alarms to the list, providing you with the most recent alarms. Once an alarm is handled, it is removed from the Active Alarm list.

The Alarm History view displays information about assignments and ownership of alarms. You can track alarms received by AMS and verify their handling status from this view.

The Alarm Summary view is a graphical representation of all the alarms that you have received.

Managing the Alarm Management System

The ZfS ConsoleOne provides a central location for monitoring, managing, and controlling critical events on your network. You can configure AMS to alert you to errors on critical systems and events to assist you in maintaining your network. This section contains the following information:

- ◆ [“Recognizing Alarm Indicators” on page 109](#)
- ◆ [“Viewing Alarms” on page 109](#)
- ◆ [“Enabling and Disabling Alarms” on page 113](#)
- ◆ [“Resolving Alarms” on page 114](#)
- ◆ [“Deleting Alarms” on page 116](#)
- ◆ [“Performing Actions on Alarms” on page 118](#)

Recognizing Alarm Indicators

You can monitor the network for alarm-triggering events by observing nodes on topology maps or Atlas views, [Active Alarm](#), and [Alarm History](#) views and in the server/node summary. The following table lists the alarm indicators and the type of alarm they are associated with.

Alarm Indicator	Applies To
Alarm icons anchored to the affected object	Alarms with severe, major, and minor severity are displayed in the Atlas and Console views and the left pane of ConsoleOne. An alarm icon remains anchored to a segment or device object until you handle all alarms outstanding against that object. Alarm icons differ based on the severity level of the alarm. See “Interpreting Alarms” on page 111 for details on alarm severity and the associated icons. Keep in mind that if a segment or device has multiple alarms logged against it, the alarm icon always depicts the highest level of severity.
Ticker-tape message on the status bar	AMS can automatically display alarm messages on the status bar. By default, this option is enabled. You can configure each individual alarm disposition to disable display of the ticker-tape message. Upon recognizing an alarm-triggering event, AMS displays a message in the status bar describing the alarm. For information on setting this option, see “Displaying a Ticker-Tape Message” on page 121 .
Audible beep	AMS can be configured to produce an audible beep on ConsoleOne when an alarm occurs. By default, this option is disabled. You can configure each individual alarm disposition to enable the audible notification. For information on setting this option, see “Making an Audible Beep” on page 122 .

Viewing Alarms

You can access active and historical alarm data from any ConsoleOne location. As an administrator, you can define access restrictions to alarm data and management functions through the role-based services to further define the data presented based on the roles in your organization.

You can modify the presentation of the alarm data displayed in the Active Alarms and Alarm History view by filtering the displayed data, changing the column layout, and changing the sorting order. All options for changing the presentation are under the View menu in ConsoleOne.

The following sections describes the different ways you can view and use alarms:

- ◆ [“Viewing Active Alarms” on page 110](#)
- ◆ [“Viewing Historical Alarms” on page 110](#)
- ◆ [“Viewing the Alarm Summary” on page 110](#)
- ◆ [“Interpreting Alarms” on page 111](#)
- ◆ [“Sorting Alarms” on page 112](#)
- ◆ [“Filtering Alarms” on page 112](#)

Viewing Active Alarms

The ZfS ConsoleOne Active Alarm view displays alarm statistics for all current alarms received from segments or devices, per management domain. The Summary view shows a list of all active alarms for that server or node.

The Active Alarms view and Server Summary view display a table of detailed information about active alarms. These views are updated whenever a new alarm occurs and is archived on your network. New alarms are appended to the list.

To display the Active Alarm view:

- 1** Click the ZENworks for Servers site object in the left frame of ConsoleOne.
- 2** Click View > Active Alarms.

The Active Alarm view is displayed. You can perform the following activities from this view:

- ◆ [“Assigning Alarms” on page 114](#)
- ◆ [“Owning Alarms” on page 115](#)
- ◆ [“Handling Alarms” on page 115](#)
- ◆ [“Adding Notes to Alarms” on page 115](#)

Viewing Historical Alarms

The Alarm History view displays information about all archived alarms, including the handling status of each alarm. You can access the Alarm History view only if you have been granted access through the role-based services.

To display the Alarm History view:

- 1** Click the ZENworks for Servers site object in the left frame of ConsoleOne.
- 2** Click View > Alarm History.

The Active Alarm view is displayed. You can perform the following alarm handling activities from this view:

- ◆ [“Assigning Alarms” on page 114](#)
- ◆ [“Owning Alarms” on page 115](#)
- ◆ [“Deleting Alarms” on page 116](#)

- ◆ “Adding Notes to Alarms” on page 115

Viewing the Alarm Summary

The Alarm Summary is a graphical representation of the summary of alarms you have received. The view is divided into three panels of representation: pie chart panel, bar graph panel, and trend panel. You can choose to view the information in these panels for a given period of time. The time duration ranges for the hour, for the day, for the week, and for the month.

- ◆ The pie chart panel includes alarm distribution based on severity, category, owner and alarm state
- ◆ The bar graph panel includes the Top N Alarm types, Top N Source Address and Top N Affected Node. The value of N is configurable. The trend displays the rate at which the alarms are received.

You can customize the pie chart and the bar graph representations to reflect the customized data.

To display the Alarm Summary view:

- 1** Click the ZENworks for Servers site object in the left frame of ConsoleOne.
- 2** Click View > Alarm Summary.

The Alarm Summary view displayed.

To customize the pie chart and the bar graph representation:

- 2a** Click the Customize button on the Alarm Summary view.

The Customize Summary view dialog box is displayed.

By default, all the options in this dialog box are selected. However, you can select the required options to customize the view.

Interpreting Alarms

The Active Alarm and Alarm History views display lists of alarms that have been archived in the alarm manager database. The alarms are displayed as a tabular list. The following table describes the data type and contents:

Data Type (Column)	Contents
Severity	Alarm icon that indicates the severity level attributed to the trap. The color of the alarm icon indicates the level of alarm severity, as follows: <ul style="list-style-type: none">  Red = Severe  Magenta = Major  Yellow = Minor  Blue = Informational  White = Unknown
From	Network address of the device that sent the alarm to AMS.

Data Type (Column)	Contents
Summary	Summary of the event, often including the name or address of the object affected by the alarm.
Owner	Person or group responsible for handling the alarm. The default owner is SYSTEM.
Received Time	Date and time when the AMS received the alarm.
Type	Generic description of the alarm. For example, volume out of disk space.
Category	Category identified in the MIB associated with the trap-type object.

You can filter the data displayed in the alarm views based on criteria from statistics displayed in each view; see [“Filtering Alarms” on page 112](#) for details. After selecting one or more alarm entries in an alarm view, you can perform operations by right-clicking them.

Sorting Alarms

You can modify the order in which the alarms are displayed on the Active Alarm or Alarm History views by sorting the alarms. By default, the alarms are sorted in ascending order by received time.

To edit the sort settings:

- 1** Click View > Settings > Sort.
- 2** Select the criteria by which you want the alarms sorted. You can sort by
 - ◆ Type
 - ◆ Severity
 - ◆ Category
 - ◆ Received time
 - ◆ Summary
 - ◆ Owner
 - ◆ Affected Object
- 3** Indicate whether you want the alarms sorted in ascending (oldest first) or descending (the most recent alarms first) order by selecting the appropriate radio button from the Sort Order box.
- 4** Click OK.

The alarms are now sorted according to the criteria you specified.

Filtering Alarms

You can display the alarms in a tabular view based on filter conditions. The filter applies only to the current management session and clears when you ConsoleOne.

You set up a filter by selecting criteria from four drop-down lists. You can either set up simple filters that require only one line, or complex filters composed of multiple lines or groups of lines. If you set up a filter using more than one line, you must also specify the logical relationship between the line and/or group of lines.

To set up a filter:

- 1** Go to the view you want to filter.
- 2** Click View > Settings > Filter.

The Alarm Filter dialog box is displayed.
- 3** Select the column by which you want AMS to filter alarms from the first drop-down list. You can filter alarms using the following columns:
 - ♦ **Severity:** Filters the alarms based on the alarm severity. Alarm severity is assigned to an alarm type.
 - ♦ **Type:** Filters alarms based on the alarm type. The alarm type is set by the SNMP trap-type defined in the MIB or the proprietary alarm definition.
 - ♦ **Category:** Filters alarms based on the category of the alarm. Alarm categories are based on the MIB that defines the trap-type objects.
 - ♦ **Generator Type:** Filters alarms based on the type of agent or system generating the alarms.
- 4** Select an operator from the second drop-down list.

The operator defines how to constrain the column you have selected to a value. For example, you can specify that the selected category must be equal to, not equal to, greater than, less than, greater than or equal to, less than or equal to, contain, or start with the value you select in the third drop-down list in order for an alarm to be displayed. Keep in mind that the list of available operators depends on what column you've selected.
- 5** Select a value from the third drop-down list.
- 6** Specify how this filter statement relates to other statements you plan to define by selecting a value from the fourth drop-down list.
 - ♦ If this is the only filter statement or if it is the last statement in a group, select End.
 - ♦ If you want to add a line below the current filter statement, select New Row. A new line is added. You must define the logical relationship between the previous line and the new line. The alarms will be displayed based on the logical condition you have specified. Select And to satisfy both the filter conditions. Select Or to satisfy any one of the filter conditions for the alarm to be displayed.
 - ♦ If you want to add one or more lines that are unrelated to the preceding lines, select New Group. A new line is added. An additional drop-down list separates the new line from the preceding lines. Select a value from this drop-down list to indicate the relations between the filter statements. Select And if you want both the filter statements to be satisfied. Select Or if you want only one of the filter statements in one of the groups to be satisfied. Select End from the fourth drop-down list when you add a new group.
- 7** Click OK if you have defined filters.

The alarm list is updated to display only those alarms that meet the filter criteria you defined.

Enabling and Disabling Alarms

ZfS provides default threshold values for managed NetWare and Windows* NT* servers and network segments hosting the Traffic Analysis Agents for a station connected to a segment. An alarm is generated if the values exceed the threshold values. The server threshold alarms are enabled by default while the segment threshold alarms are not. You will need to enable threshold alarms to receive.

IMPORTANT: In order to modify the segment properties, you must have the Traffic Analysis Agents for NetWare or Windows NT hosted on a station, connected to the segment.

To enable or disable segment threshold alarms:

- 1** Right-click the segment object > click Properties.
- 2** If it is not already displayed, select the Segment Alarms tab.
- 3** Select the alarm you want to enable or disable > click Edit.
- 4** In the Value field, enter the threshold value after which an alarm should be generated.
- 5** Enter the time (in seconds) that the threshold value must exceed in order to generate an alarm in the Sampling Interval field.
- 6** Check the Enable check box.
- 7** Click OK.

Resolving Alarms

Alarms that occur on segments and devices on your network are added to the alarm manager database and are presented in the Active Alarms and Alarm History views. Entries in the alarm manager database remain in the database until the alarm is deleted. The database records the status of the alarm from first acknowledging the alarm, assigning it to a group or user, owning the alarm, and finally deleting it from the database once the owner has resolved the problem.

Resolution operations for alarms are displayed when you right-click a single entry or multiple entries in an alarm view and click any of the following actions:

- ◆ [“Assigning Alarms” on page 114](#)
- ◆ [“Owning Alarms” on page 115](#)
- ◆ [“Handling Alarms” on page 115](#)
- ◆ [“Adding Notes to Alarms” on page 115](#)
- ◆ [“Jump to the Affected Node” on page 116](#)

You can also access the alarm action menu items from the View menu in ConsoleOne.

The order in which you perform the handling, assigning, and owning of an alarm or multiple alarms depends on your organization. Keep in mind that after you handle an alarm, it is removed from the Active Alarms list and only appears in the Alarm History list. A suggested course for resolving an alarm is for you to first assign the alarm to a group or team member, then have someone from the group take ownership of the alarm. When the network problem or event has been resolved, the team member can handle the alarm to remove it from the Active Alarms list. By following this process, you can track the alarm status through resolution, and finally delete the alarm from the Alarm History list.

Assigning Alarms

You can specify the group or user that is assigned to handle an alarm. This allows you to use any team assignments you already have within your organization. For example, you may have a group or team member assigned to handle all alarms relating to NetWare servers. You can assign one or more alarms to a group or user. Note, however, that you must have been granted access to assign alarms through the role-based services. You can use an alarm filter to help you determine groups based on certain filtering criteria. See [“Filtering Alarms” on page 112](#) for information on filtering options.

TIP: This is optional and is provided for tracking the status of alarm resolution.

To assign an alarm:

- 1** Select the alarm you want to assign from the **Active Alarm** or **Alarm History** list.
- 2** Click View > Assign.
- 3** Enter the name of the person or group to which you want to assign the alarm in the Username field.

The name you enter does not correlate to users in eDirectory and can represent the organization structure you already have in place.

- 4** Click OK.

Owning Alarms

A user can take ownership of one or more alarms. If a user is a member of a group assigned to resolve a network problem, the team member can take ownership of the alarm and finally delete the alarm to remove it from the alarm manager database.

TIP: This is optional and is provided for tracking the status of alarm resolution.

To take ownership of an alarm:

- 1** Select the alarm from the **Active Alarm** or **Alarm History** view.
- 2** Click View > Own.

The value in the Owner field changes to the eDirectory name you are logged in as. Note that you cannot customize this option; the user logged in to ConsoleOne will always become the owner of the alarm when this action is used.

Handling Alarms

Alarms displayed in the Active Alarm view have not been handled by anyone. After the alarm is handled, it is removed from the Active Alarm list, and any alarm indicators shown in other views in ConsoleOne are removed. See **“Recognizing Alarm Indicators” on page 109** for information on different types of alarm indicators. Note that the alarm is still displayed in the **Alarm History** view.

To handle an alarm:

- 1** Select the alarm from the **Active Alarm** list.
- 2** Click View > Handle.

The alarm is removed from the Active Alarm list. You can still display information about the alarm by switching to the **Alarm History** view.

Adding Notes to Alarms

You can add a note to any of the alarms displayed in the Active Alarm view or Alarm History view. The note can contain any relevant useful information about the alarm.

To handle an alarm:

- 1** Select the alarm from the **Active Alarm** or **Alarm History**.
- 2** Click View > Note.

The Note dialog box is displayed.

Create a note for the alarm.

3 Click OK.

The alarm icon will now have a note icon associated with it, indicating that a note has been added to the alarm.

If you want to delete the note from the alarm, repeat step 2. Delete the note that you created in the Note dialog box.

Click Apply. The note will be deleted for the alarm, and the note icon will not be displayed.

Jump to the Affected Node

You can jump to the affected node where the alarm has been triggered and perform the necessary action to rectify the affected node.

To jump to the affected node alarm:

1 Select the alarm from the **Active Alarm** or **Alarm History**.

2 Click View > Jump to Affected Node.

The Console view is displayed and the node on which the alarm has triggered is highlighted.

Deleting Alarms

Alarms displayed in the Alarm History view can be deleted from the alarm list after problem resolution. You can delete one or more alarm entries to remove the alarm from the list. Note that to delete an alarm, you must have been granted access to view alarm history and to delete alarms through the role-based services.

There are two ways to delete alarms:

- ◆ You can delete alarms manually from the Alarm History view. See “[Deleting Alarms from ConsoleOne](#)” on page 116.
- ◆ You can delete alarms automatically using the AMS purge utility. See “[Deleting Alarms Using the Purge Utility](#)” on page 116.

IMPORTANT: The alarm manager database, located on the management server, records the status of every alarm instance received by the AMS. You must be diligent in deleting alarms after a problem is resolved in order to keep the database from taking up excessive disk space. Currently, the alarm manager database uses the Alarm purge utility (on by default) to automatically delete entries after a period of time or based on the size of the database.

Deleting Alarms from ConsoleOne

You can manually delete alarms from ConsoleOne.

To delete alarms:

1 Select the alarms you want to delete from the **Alarm History** list.

2 Click View > Delete.

The alarms are removed from the Alarm History view.

Deleting Alarms Using the Purge Utility

You can delete alarms automatically using the AMS purge utility. Before you can use this utility, you must set up the utility’s configuration file, AMPURGE.PROPERTIES, which is located in the properties directory on the server and volume where you installed the alarm manager database.

Then you can schedule the utility to run automatically at a specified time of day. Or, you can run the utility manually from the server console. The following sections describe how to set up and use the AMS purge utility:

- ♦ [“Setting Up the Purge Utility Configuration File” on page 117](#)
- ♦ [“Setting Up the Purge Utility to Run Automatically” on page 117](#)

Setting Up the Purge Utility Configuration File

The AMS purge utility configuration file, AMPURGE.PROPERTIES, defines the criteria for selecting the alarms to be purged as well as the time of day the process should run. This file is located in the properties directory on the server and volume where you installed the alarm manager database.

Before you can run the purge utility, you must set up the configuration file as follows:

- 1** Open the AMPURGE.PROPERTIES file with a text editor.
- 2** Set the criteria for purging alarms by editing the values of the following lines in the file:
 - ♦ `SeverityInformationalPurgeWait`: The number of days before informational alarms will be purged.
 - ♦ `SeverityMinorPurgeWait`: The number of days before minor alarms will be purged.
 - ♦ `SeverityMajorPurgeWait`: The number of days before major alarms will be purged.
 - ♦ `SeverityCriticalPurgeWait`: The number of days before critical alarms will be purged.
 - ♦ `SeverityUnknownPurgeWait`: The number of days before unknown alarms will be purged.

By default, alarms of all severity levels are purged after seven days.

- 3** Save the configuration file.

Setting Up the Purge Utility to Run Automatically

You can schedule the purge utility to run daily to ensure that the alarm manager database does not consume excessive disk space. Before you can set up the utility to run automatically, you must make sure to set up the file with your preferences for deleting alarms of various severities. See [“Setting Up the Purge Utility Configuration File” on page 117](#).

To set up the utility to run automatically:

- 1** Open the AMPURGE.PROPERTIES file with a text editor.
- 2** Set the time of day you want the utility to run by editing the `PurgeStartTime` entry.

Valid values are 0 to 23, where 0 is midnight and 23 is 11:00 p.m. Keep in mind that the purge utility is memory intensive and can occupy the server for several minutes. Therefore, you should set the utility to run during off-peak hours.
- 3** Save and close the file.
- 4** Open the ALARMMANAGER.PROPERTIES file and verify that the following line exists:
`AlarmPurgeService=yes`

If the line does not exist, add it to the end of the file.
- 5** Save and close the file.

- 6 Restart the server.

Performing Actions on Alarms

You can configure an alarm to automatically perform an action when an alarm occurs. You do this by editing the alarm dispositions associated with each alarm template. Alarm dispositions are created for each alarm template in the Alarm Manager database and default settings are assigned. You can edit the alarm dispositions to enable the following actions:

- ◆ “Sending SMTP Mail Notification” on page 118
- ◆ “Launching an External Program” on page 119
- ◆ “Forwarding SNMP Traps to Other Management Systems” on page 120
- ◆ “Forwarding Alarms to Other Management Servers” on page 121
- ◆ “Displaying a Ticker-Tape Message” on page 121
- ◆ “Making an Audible Beep” on page 122
- ◆ “Archiving Alarm Statistics” on page 122
- ◆ “Sorting Alarm Templates” on page 123

Sending SMTP Mail Notification

You can send SMTP messages to recipients who are specified to receive e-mail notification.

To modify alarm disposition to automatically send SMTP mail notification:

- 1 Right-click the ZENworks for Servers site object in the left frame of ConsoleOne > click Properties.
- 2 Click the Alarm Disposition tab.
- 3 Select the alarms you want to edit from the Alarm Templates list > click Edit.
The Edit Alarm Disposition dialog box is displayed.
- 4 Click the SMTP Mail Notification tab.
- 5 Check the Notify through SMTP Mail check box.
- 6 Enter the IP address of the SMTP host server that handles incoming and outgoing e-mail in the SMTP Host field.
- 7 Enter the name of the person sending the notification in the From field.
- 8 Enter the e-mail addresses of the recipients in the To field.
- 9 Enter the subject of the e-mail in the Subject field.
- 10 Enter a message for the e-mail, if any, in the Message field.
- 11 Click OK.

Note that the subject and message text strings can contain any of the variables listed in the following table. These variables allow you to add details to your message about the segment or device generating the fault or event. All variables must be preceded by a percent sign (%). For example, the subject line could include the %v variable to display the severity of the alarm. You can also specify the width for the variables. %(*nnn*)X can be used to limit the length of the %X value to *nnn* characters. X represents any format specifier. For example, %(10)a will display the Alarm ID up to 10 characters.

Variable Parameter	Name	Description
a	Alarm ID	Identification number of the alarm as it is stored in the database.
c	Affected class	Class of equipment that sent the alarm. This can be any portion of the network and is categorized in the database for indexing.
o	Affected object number	Identification number of the node that generated the alarm as it is stored in the database.
s	Alarm summary string	Message describing the alarm. (This is the same as the status bar ticker-tape message.)
t	Alarm type string	Description of the alarm. This matches the description in the Alarm Type column in the Alarm Summary window.
v	Severity number	Alarm severity can be 1 = severe 2 = major 3 = minor 4 = informational All others are unknown.
n	Affected object name	Identification name of the node affected by the alarm.
p	Source Address	The source address of the agent that generated the alarm.
-h	Remove Default Header	Truncates the default header while sending an SMTP message.

Launching an External Program

As part of editing the disposition of an alarm, you can set options to launch any program on the ZfS server automatically when an alarm is received. For example, you might want an alarm to launch a program that sends a message to the system administrator's pager.

In addition to specifying the program to launch, you can also specify arguments and variables to be passed to the program.

Although ZfS provides the capability to launch applications, the product does not supply any predefined programs. However, you can launch an NLM and run scripting routines or use third-party programs.

To set up automatic application launching:

- 1** Right-click the ZENworks for Servers site object in the left frame of ConsoleOne > click Properties.
- 2** Click the Alarm Disposition tab.
- 3** Select the alarm that you want to edit from the Alarm Templates list > click Edit.
The Edit Alarm Disposition dialog box is displayed.
- 4** Click the Launching Application tab.

- 5** Check the Launch Application check box.
- 6** Enter the path and name of the application in the Application Name field.
- 7** Enter any necessary execution arguments or script variables in the Argument field > click OK.

Arguments are passed directly to the program; text is not parsed, but is read as literal text strings. Variables must be preceded with a percent sign (%). The percent sign can be followed by an optional length field that limits the length to which the parameter can expand. You can also specify the width for the variables. `%(nnn)X` can be used to limit the length of the `%X` value to *nnn* characters. X represents any format specifier. For example, `%(10)a` will display the Alarm ID up to 10 characters.

The following table lists the variables you can use when launching a program.

Variable	Name	Description
a	Alarm ID	Identification number of the alarm as it is stored in the database.
c	Affected class	Class of equipment that sent the alarm. This can be any portion of the network and is categorized in the database for indexing.
o	Affected object number	Identification number of the node that generated the alarm as it is stored in the database.
s	Alarm summary string	Message describing the alarm. (This is the same as the status bar ticker-tape message.)
t	Alarm type string	Description of the alarm. This matches the description in the Alarm Type column in the Alarm Summary window.
n	Affected object name	Identification name of the node affected by the alarm.
p	Source Address	The source address of the agent that generated the alarm.
v	Severity number	Alarm severity can be 1 = severe 2 = major 3 = minor 4 = informational All others are unknown.

Forwarding SNMP Traps to Other Management Systems

AMS can be configured to forward an unmodified SNMP trap. Specify the IP address of the target management station or server in the alarm disposition and the trap is automatically forwarded.

To forward SNMP traps:

- 1** Right-click the ZENworks for Servers site object in the left frame of ConsoleOne > click Properties.
- 2** Click the Alarm Disposition tab.
- 3** Select the alarm that you want to edit from the Alarm Templates list > click Edit.

The Edit Alarm Disposition dialog box is displayed.

- 4** Click the SNMP Trap Forwarding tab.
- 5** Enter the IP address of the server to which you want to forward traps in the SNMP Target Address field > click Add.

The server is added to the List of Targets. Repeat this step for all servers you want to receive the traps.
- 6** Click OK.

Forwarding Alarms to Other Management Servers

AMS can be configured to forward a processed alarm to other ZfS management servers. You specify the IP address or server name of the target management server in the alarm disposition and the alarm is automatically forwarded.

To forward alarms:

- 1** Right-click the ZENworks for Servers site object in the left frame of ConsoleOne > click Properties.
- 2** Click the Alarm Disposition tab.
- 3** Select the alarm that you want to edit from the Alarm Templates list > click Edit.

The Edit Alarm Disposition dialog box is displayed.
- 4** Click the Alarm Forwarding tab.
- 5** To add a target server to receive the alarms:
 - 5a** Select the ZfS site to which you want to forward alarms in the Site Name field.
 - 5b** Select the ZfS host to which you want to forward alarms in the Site Host field.
 - 5c** Click Add.

The server is added to the List of Targets. Repeat this step for all servers to which you want to forward alarms.
- 6** Click OK.

Displaying a Ticker-Tape Message

The alarm disposition includes other configuration settings that include displaying a ticker-tape message in the status bar of ConsoleOne. The message provides a summary of the most recent alarm or network event.

This option is enabled by default. You may want to edit your alarm dispositions so that only important alarms that you want to monitor display a ticker-tape message.

To disable or enable a ticker-tape message:

- 1** Right-click the ZENworks for Servers site object in the left frame of ConsoleOne > click Properties.
- 2** Click the Alarm Disposition tab.
- 3** Select the alarm that you want to edit from the Alarm Templates list > click Edit.

The Edit Alarm Disposition dialog box is displayed.
- 4** Click the Other Configuration tab.
- 5** To disable the ticker-tape message, uncheck the Show on Ticker Bar check box.

or

To enable the ticker-tape message, check the Show on Ticker Bar check box.

6 Click OK.

Making an Audible Beep

The alarm disposition includes other configuration settings that include making an audible beep at ConsoleOne. The sound alerts the user of an occurrence of an alarm. Useful applications of this function include:

- ◆ Serverabend
- ◆ System: Server downed by user
- ◆ File system full

This option is disabled by default. You should enable this option for important alarms that you want to monitor.

To enable or disable an audible beep:

- 1** Right-click the ZENworks for Servers site object in the left frame of ConsoleOne > click Properties.
- 2** Click the Alarm Disposition tab.
- 3** Select the alarm that you want to edit from the Alarm Templates list > click Edit.
The Edit Alarm Disposition dialog box is displayed.
- 4** Click the Other Configuration tab.
- 5** To enable the audible beep function, check the Beep on Console check box.

or

To disable the audible beep function, uncheck the Beep on Console check box.

6 Click OK.

Archiving Alarm Statistics

The AMS system provides data to the reporting tools to generate detailed reports on alarms and network events. Enabling the Archive option stores the alarm in the alarm manager database on the management server. This option is enabled by default. You should disable this option only on the types of alarms that you do not want to track and analyze.

To enable or disable alarm archiving:

- 1** Right-click the ZENworks for Servers site object in the left frame of ConsoleOne > click Properties.
- 2** Click the Alarm Disposition tab.
- 3** Select the alarm that you want to edit from the Alarm Templates list > click Edit.
The Edit Alarm Disposition dialog box is displayed.
- 4** Click the Other Configuration tab.
- 5** To disable alarm archiving, uncheck the Archive check box.

or

To enable alarm archiving, check the Archive check box.

- 6 Click OK.

Sorting Alarm Templates

The AMS system enables you to sort the alarm templates based on different conditions. This option is enabled by default. You can sort the templates based on Severity, Generator Type, Category or Type. By default, the sorting is done based on the Type. You can also sort the templates based on a single field by selecting the field from the drop-down list under the Sort Items By option, or you can sort the templates based on different combinations of fields by using the Then By options.

To sort the alarm templates:

- 1 Right-click the ZENworks for Servers site object in the left frame of ConsoleOne > click Properties.
- 2 Click the Alarm Disposition tab.
- 3 Click the Sort button.

The Template Sorting dialog box is displayed.

- 4 Select fields from Sort Items By drop-down list.
- 5 Select fields from Then By drop-down list.
- 6 Select fields from Items by drop-down list.
- 7 Click OK.

The templates are sorted based on the field selected in the Sort Items By option and the fields selected in these options. For example, if you have chosen to sort the templates based on Severity in the Sort Items By list, and Category, Generator Type, in the three Then By lists, the templates will be sorted first based on severity, then on the category, followed by the generator type and the type of the template.

Maintaining the Alarm Management System

The alarm manager database on the ZfS management server increases in size each time AMS logs an alarm.

IMPORTANT: If you do not control the size of this database, it can increase until it fills the hard disk on the management server.

To control the size of the alarm manager database, regularly delete alarms that have been resolved or alarms that are not required for future reference or action. This deletes the instance of the alarm record from the alarm manager database and thus controls the size of the database.

You can delete alarms from the Alarm History view in ConsoleOne under the View menu. For more information, see [“Deleting Alarms” on page 116](#).

Troubleshooting the Alarm Management System

When AMS receives an unsolicited SNMP trap from an agent, it locates the appropriate alarm template for the trap-type object that is defined in the MIB of the device. If the alarm template is not available, the AMS checks the IgnoreUnknownTrap flag in the
<install_volume>\<install_dir>\ZENWorks\MMS\MWServer\Properties\Alarmmanager.properties

ies file. If the flag value is set to True the alarm is ignored. If the flag value is set to False the alarm is archived in the database as an unknown trap. If the flag value is set to Yes the alarm is ignored. If the flag value is set to No the alarm is archived in the database as an unknown trap. The default value of the flag is set to Yes.

To resolve this problem you need to add the MIB of the device to the MIB Pool on the management server. The MIB contains the trap definitions for traps sent from the device. If the trap-type object is undefined by AMS, it cannot resolve the type of alarm received from the trap object identifier (OID), and the alarm is unknown. See [Chapter 6, “Using the MIB Tools,” on page 165](#) for information on compiling MIBs and adding MIBs to the MIB Pool.

If you add a new device to your network, you must add the MIB to the MIB Pool. If the SNMP agent is a proxy agent hosted on a station and the software is updated, you need to update the MIB in the MIB Pool.

5

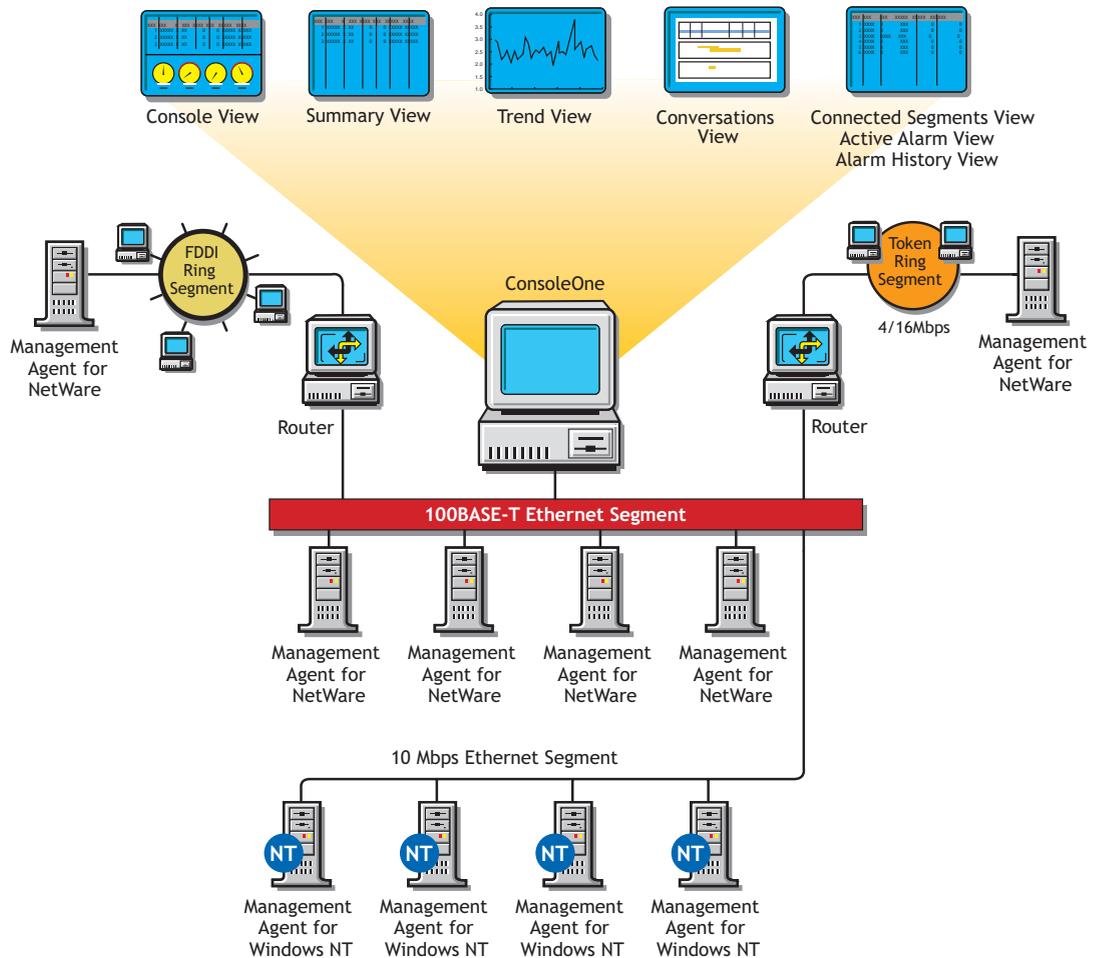
Understanding Server Management

The ZENworks[®] for Servers (ZfS) Server Management components allow you to monitor, configure, and control the managed servers and nodes on your network. The SNMP-based server Management Agents for NetWare[®] and Windows* NT* servers provide real-time server performance data and information about server alarms and events to the network management console. By selecting a server or node from atlas page maps or hierarchical lists in the left pane of ConsoleOne, you can access three main views of information:

- ♦ **Console View:** Provides details about the selected server or node. You can drill down into the server configuration to display information about the internal components of the machine, such as the devices, operating system, and services available on the machine.
- ♦ **Summary View:** Provides details about the server performance, such as alarms generated by the server, CPU utilization, and available disk space. By drilling down into the server configuration, you can also view summary information about other components, such as processors, threads, memory, and volumes.
- ♦ **Trend View:** Displays graphical representations of trend parameters, allowing you to monitor the state of a server over various periods of time. Using trend data, you can track the health status of servers, allowing you to predict potential problems and plan for future expansion of server configurations.

In addition to viewing information about the servers on your network, the server management components also enable you to configure your managed NetWare servers and execute frequently used commands from ConsoleOne.

The following figure displays a functional view of the ZfS Server Management components. It illustrates the Management Agent for NetWare and Management Agent for Windows NT distributed throughout a network.



This section contains the following topics to help you understand the server management components:

- ◆ [“Understanding Server Management” on page 126](#)
- ◆ [“Planning for Server Management” on page 128](#)
- ◆ [“Optimizing Server Management” on page 130](#)
- ◆ [“Managing Servers” on page 137](#)
- ◆ [“Object Hierarchy and View Details” on page 144](#)

Understanding Server Management

The Management Agent for NetWare and the Management Agent for Windows NT include features that offer benefits over server management functionality included with NetWare and Windows NT server software.

This section includes the following topics:

- ◆ [“SNMP-Based Server Management” on page 127](#)
- ◆ [“SNMP Agent Functions” on page 127](#)

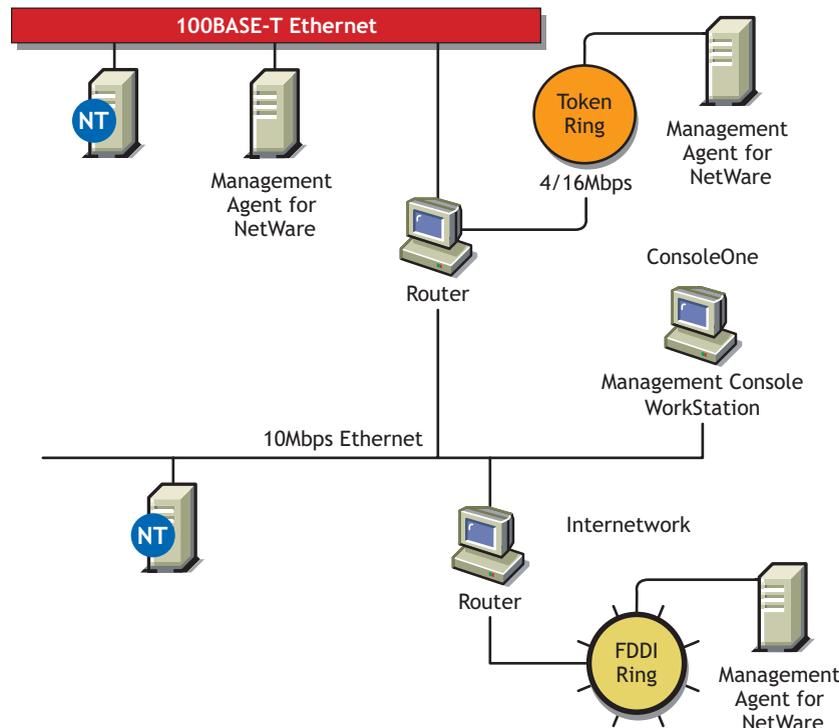
SNMP-Based Server Management

The main advantage of the Management Agent for NetWare and Management Agent for Windows NT is that they support the industry standard Simple Network Management Protocol (SNMP). SNMP is the protocol governing network management and the monitoring of network devices and their functions.

The ZfS SNMP agents support UDP/IP, IPX™, and NCP implementations for accepting and sending packets (datagrams). This standard mechanism allows any SNMP console or manager to request information from the ZfS Server Management SNMP agents. An SNMP console can be any console that supports SNMP; the ZfS ConsoleOne fully supports SNMP v.1 communication.

The ZfS server management SNMP agents run on NetWare and Windows NT servers in your network. The agents monitor servers, collecting historical data and dynamic data in response to requests from ConsoleOne. An administrator at the ZfS ConsoleOne can request data simply by clicking a representative icon for any device, operating system, or service discovered on a server.

The following figure illustrates an internetwork using the Management Agent for NetWare and Management Agent for Windows NT and the ZfS ConsoleOne.



SNMP Agent Functions

The functionality of the Management Agent for NetWare and Management Agent for Windows NT (the Novell SNMP-based agents for NetWare and Windows NT servers) can be divided into the following areas:

- ◆ Collecting Statistics
 - ◆ Monitoring: Server monitoring provides instant information about various monitored elements of the server, such as CPU utilization, memory size, cache buffers, connected users, volumes, disks, disk space usage per user, network adapters, print queues, print jobs, and loaded NetWare Loadable Module™ (NLM™) files on NetWare or Windows NT servers.
 - ◆ Trending: Trends provide historical data about various server objects and can be displayed in a diagram on the SNMP console. Trends are stored at the server side, which eliminates the need for continuous polling from an SNMP manager, and this data can be accessed via SNMP by any ZfS ConsoleOne or other SNMP-based console.
- ◆ Alarm Notification: More than 580 different types of alarms or events (SNMP traps) can be sent from any NetWare server to the ZfS management system or to any other SNMP-based console.

Any Windows NT system, security, or application event is converted to an SNMP trap and sent to the ZfS management system or to any other SNMP-based console.

The alarms inform the administrator about events that have occurred or thresholds which have been crossed.

- ◆ Configuration Management: The Management Agent for NetWare enables network administrators to remotely configure NetWare servers. There are 187 SET parameters on the NetWare server that can be used to tune the server's performance. Administrators can view settings and change all parameters from any ZfS ConsoleOne.

The SNMP agents must be installed on any server that you want to manage. For information on installing the SNMP agents, or if you have already installed the agent software to servers that you want to manage, see [“Installing and Setting Up Management and Monitoring Services”](#) in the *Installation* guide.

Planning for Server Management

A baseline defines the typical activity of your network servers. Keeping a baseline document of activity on a server lets you determine when the activity is atypical. To create a baseline activity, you should gather statistical information when the server is functioning typically.

This section contains the following information to help you plan your server management strategy:

- ◆ [“Creating a Baseline of Typical Server Activity”](#) on page 128
- ◆ [“Using the Baseline Document”](#) on page 129
- ◆ [“Server Baseline Document Tips”](#) on page 129

Creating a Baseline of Typical Server Activity

For server statistics such as CPU utilization, you should create a trend graph that plots information over a period of time. Statistics sampling that gathers data over a short period of time can be

misleading. If you modify the server's configuration, it is useful to create another baseline against which you can compare future activity.

There are two ways to create baseline documents. The first is to create them manually by printing the various trend graphs for which you want to maintain baselines. The other way is to use the server management health reports as your baseline documents. For more information on creating and generating health reports, see [“Managing the Server Management Health Reports” on page 297](#). In either case, the data gathered can be exported into programs, such as spreadsheets, for further analysis and to maintain records over time.

Using the Baseline Document

The following sections will help you plan and use the baseline document:

- ◆ [“Using Baseline Documents to Set Alarm Thresholds Appropriately” on page 129](#)
- ◆ [“Using Baseline Documents to Track Server Utilization” on page 129](#)
- ◆ [“Use Baseline Documents in Troubleshooting” on page 129](#)

Using Baseline Documents to Set Alarm Thresholds Appropriately

You should set alarm thresholds for statistics on servers monitored by the SNMP agent software, so that if the threshold is exceeded, you are notified at ConsoleOne. Setting alarm threshold values for statistics on a server eliminates the need for you to constantly monitor polled server statistics for problems.

Server Management components provide default values for thresholds set on server statistics; rising and falling statistics generate an alarm when a threshold is surpassed.

Using Baseline Documents to Track Server Utilization

By comparing current server performance statistics against the performance recorded in your baseline document, you can determine how performance is affected by server configuration changes. This comparison also helps you plan for growth and justify upgrades and expansion. You can view graphs of real-time trends and historical trends over hourly, daily, weekly, monthly, and yearly periods.

Use Baseline Documents in Troubleshooting

By knowing what the typical server activity is, you can recognize atypical activity, which might help you isolate the cause of a problem.

Server Baseline Document Tips

You should include the following key characteristics in each server baseline document:

- ◆ [“CPU Utilization” on page 129](#)
- ◆ [“Cache Buffers” on page 130](#)
- ◆ [“File Reads and Writes” on page 130](#)
- ◆ [“Volume Utilization” on page 130](#)
- ◆ [“Running Software” on page 130](#)

CPU Utilization

The CPU Utilization statistic indicates how busy the microprocessor is. High CPU utilization can cause slow network response time. Utilization is likely to be higher at some times during the day (for example, when users log in to the network in the morning, or access e-mail), week, or month. Tracking CPU utilization helps you track the load on the server processor at peak and low times. This information helps you determine the effect of current system and application processor demands and analyze the impact on performance.

Cache Buffers

Virtually all processes are handled through server cache, a block of server memory (RAM) in which files are temporarily stored. Cache buffers greatly increase server performance and enable workstations to access data quicker because reading from and writing to memory is much faster than reading from or writing to disk. The optimum cache buffer is 65% to 75% of total server memory (more does not hinder performance). Low cache buffers can cause slow server performance and abends. Service degrades noticeably at 45% of total server memory.

File Reads and Writes

By tracking data about file reads and writes in your baseline, you might be able to determine whether a bottleneck is caused by the disk I/O channel. For example, if an increasing number of “server busy” packets are sent to users and there is also an increase in the file read and write number, the cause of the bottleneck might be a slow disk I/O channel or bad disk adapter driver.

Volume Utilization

Tracking volume utilization is primarily for capacity planning. By tracking the volume space used over time, you can accurately predict when you must purchase additional storage. Tracking volume utilization can also help you prevent the server from running out of disk space.

Running Software

By including information about running software in your baseline, it is easier to spot a problem application when comparing software on different servers. It is useful to also include the memory each application uses. Then, if the server is running short of memory, you can quickly see which applications are using the most memory.

Optimizing Server Management

Examine each of the configuration options in the sections that follow to determine whether you require any of the functionality provided:

- ◆ [“Setting Default Trends and Thresholds” on page 130](#)
- ◆ [“Controlling Alarm Generation” on page 134](#)
- ◆ [“Defining Recipients for SNMP Alarms” on page 136](#)

Setting Default Trends and Thresholds

You can modify the default trends and threshold values from within ConsoleOne or manually modify files on servers that have the Management Agent for NetWare or Management Agent for Windows NT software installed.

When server agents are first loaded, the initial (default) values for trends and thresholds are read from the NTREND.INI file (NetWare) or the N_NTREN.INI file (Windows NT). The initial values are also used whenever a new trend file is created. A new trend file is created when an instance of a monitored object (volume, disk, interface, and so on) is discovered on the server.

The following is a sample excerpt from an NTREND.INI file:

#	#Parameter	Sample Interval	Trend Buckets	Enbl	Rising	Falling	Enbl	Type
#	NUMBER_LOGGED_IN_USERS	5	60	1	100	90	1	rising
	NUMBER_LOGGED_IN_USERS	7	8928	1	90	81	1	rising
	NUMBER_CONNECTIONS	5	60	1	0	0	0	rising
	NUMBER_CONNECTIONS	7	8928	1	0	0	0	rising
	FILE_READS	5	60	1	0	0	0	rising
	FILE_READS	7	8928	1	0	0	0	rising
	FILE_WRITES	5	60	1	0	0	0	rising
	FILE_WRITES	7	8928	1	0	0	0	rising
	FILE_READ_KBYTES	5	60	1	0	0	0	rising
	FILE_READ_KBYTES	7	8928	1	0	0	0	rising
	FILE_WRITE_KBYTES	5	60	1	0	0	0	rising
	FILE_WRITE_KBYTES	7	8928	1	0	0	0	rising
	LSL_IN_PACKETS	5	60	1	0	0	0	rising
	LSL_IN_PACKETS	7	8928	1	0	0	0	rising
	LSL_OUT_PACKETS	5	60	1	0	0	0	rising
	LSL_OUT_PACKETS	7	8928	1	0	0	0	rising
	NCP_REQUESTS	5	60	1	0	0	0	rising
	NCP_REQUESTS	7	8928	1	0	0	0	rising
	CPU_UTILIZATION	5	60	1	90	81	1	rising
	CPU_UTILIZATION	7	8928	1	80	72	1	rising
	CACHE_BUFFERS	5	60	1	45	40	1	falling
	CACHE_BUFFERS	7	8928	1	0	0	1	falling
	CODE_DATA_MEMORY	5	60	1	0	0	0	rising
	CODE_DATA_MEMORY	7	8928	1	0	0	0	rising

After the Management Agent for NetWare and Management Agent for Windows NT software is running, trend and threshold values can be changed (using ConsoleOne) by making use of the threshold-setting features of ZfS. If the server is brought down, it retains the last trend and threshold settings that were set. Initial values are reset when any of the following situations occurs:

- ◆ Trend files have been deleted manually.
- ◆ If the server configuration is modified, for example, by adding a new volume, disk, or interface.

IMPORTANT: Trends are not maintained for CD volumes. Therefore, changing trend parameters for CD volumes has no effect.

The following sections contain information to help you modify initial trend and threshold values:

- ◆ [“Changing the Initial Trend Values” on page 131](#)
- ◆ [“Changing the Initial Threshold Values” on page 134](#)

Changing the Initial Trend Values

The trend values in the NTREND.INI file (NetWare) and N_NTREN.INI file (Windows NT) specify the time interval (Sample Interval) at which a particular trend parameter is sampled, the duration of time for which those samples are kept (Trend Buckets), and whether this sampling parameter is enabled (Enbl). For each value specified by a line in the NTREND.INI file or N_NTREN.INI file, a trend record is stored in a separate file in the SYS:\NTREND directory on a NetWare server and the \TRENFILE directory on a Windows NT server.

The following illustration depicts a line in the NTREND.INI file for the NUMBER_LOGGED_IN_USERS trend parameter with a Sample Interval of 5, Trend Buckets specified at 60, and the enable parameter specified at 1 (enabled).

```

#-----#
# Parameter | Sample Interval | Trend Buckets | Enbl | Threshold Rising | Falling | Enbl | Type |
#-----#
NUMBER_LOGGED_IN_USERS 5 60 1 100 90 1 rising

```

The following sections describe how to set or alter each of the parameters required for a trend file:

- ◆ “Setting the Sample Interval” on page 132
- ◆ “Setting the Trend Buckets” on page 132
- ◆ “Enabling or Disabling a Trend File” on page 133
- ◆ “Backing Up Trend Data” on page 133

You can specify more than one sampling interval or duration for any trend parameter by creating another line in the NTREND.INI file or N_NTTREN.INI file.

Setting the Sample Interval

The trending software enables you to collect samples of a specified parameter at any of 12 possible time intervals (Sample Interval), from 5 seconds to 1 day.

Each of these sample intervals is specified by a code number in the NTREND.INI file and the N_NTTREN.INI file. The following table specifies the codes used in the NTREND.INI and N_NTTREN.INI files for the permitted sample intervals. For example, if you want to sample a particular trend parameter once every hour, you would use the code 9.

Sample Interval	Code
5 seconds	1
10 seconds	2
15 seconds	3
30 seconds	4
1 minute	5
5 minutes	6
15 minutes	7
30 minutes	8
1 hour	9
4 hours	10
8 hours	11
1 day	12

Setting the Trend Buckets

After you have determined a sample interval for collecting samples, you must set a duration of time for which you want to collect samples. For example, if you selected a sample interval of one hour for a particular parameter, you might decide that you want to be able to review the state of that parameter for every hour over the duration of a day.

You determine the duration of time for which a parameter is collected by the number of trend buckets you specify. You must specify a trend bucket for each sample that is collected over a specific period of time. For example, to review the state every hour for 1 day, 24 trend buckets (1 per hour x 24 hours in a day) are required.

The number of trend buckets required for any particular time duration and sample interval is calculated easily. However, for your convenience, the following table shows the number of trend buckets required for each sample interval allowed, for each of seven possible time durations of from 1 hour to 1 year.

After you set the sample interval and the time duration for trend collection, you can compute the size of trend files. The number of trend buckets possible, and the approximate size in kilobytes (in parentheses), for a given sample interval and time duration are also given in the following table. The size of each trend bucket is 4 bytes plus 512 bytes for the header file. For example, if the sampling interval is 5 seconds for a period of 1 hour, the file size would be 720 trend buckets x 4 bytes long (rounded to the closest 4 KB boundary) plus 512 bytes for a total of 4.5 KB. There are always as many trend files as there are enabled trends.

Sample Interval	1 Hour Duration	1 Day Duration	1 Week Duration	1 Month Duration	3 Months Duration
0 seconds	720	17280	120960	535680	1607040
1.0 seconds	360	8640	60480	267840	803520
15 seconds	240	5760	40320	178560	535680
30 seconds	120	2880	20160	89280	267840
1 minute	60	1440	10080	44640	133920
5 minutes	12	288	2016	8929	26784
15 minutes	4	96	672	2975	8928
30 minutes	2	48	336	1488	4464
1 hour	1	24	168	744	2232
4 hours		6	42	186	558
8 hours		3	21	93	279
1 Day		1	7	31	93

After a particular time duration is exceeded for a file (all the trend buckets have been filled), the oldest samples are overwritten by the most recent samples. This means that the file contains the most recent duration recorded. For example, if you select a sample interval of 1 hour for a duration of 24 hours (using 24 trend buckets), the associated file contains the trend data for the last 24 hours.

Enabling or Disabling a Trend File

Each line in the NTREND.INI file and the N_NTTREN.INI file contains a parameter that either enables or disables the trending value to begin creating a trend file at startup. To enable the collection of data for a trend file, set this parameter to 1. To disable the collection of data for a trend file at startup, set this parameter to 0.

Backing Up Trend Data

Trend data is not automatically backed up. If you want to back up this data, you must do so manually.

Changing the Initial Threshold Values

The default threshold values in the NTREND.INI file and the N_NTTREN.INI file specify when a trap is generated. User-defined values are stored in the trend file header. If the parameter rises above or falls below the set threshold value, a rising or falling trap type is sent.

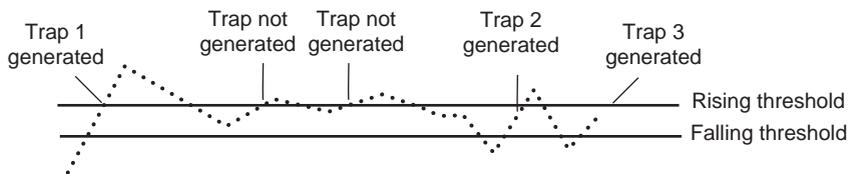
The following sections describe how to set or alter each of the parameters required for a threshold value:

- ◆ [“Setting Rising and Falling Thresholds” on page 134](#)
- ◆ [“Enabling or Disabling a Threshold Trap” on page 134](#)

Setting Rising and Falling Thresholds

Each line in the NTREND.INI file and the N_NTTREN.INI file contains a parameter for the rising threshold and the falling threshold. For each sample interval, a rising or falling trap can be generated as specified. After a trap is generated, another such trap is not generated until the sampled value falls below this threshold and reaches the falling threshold.

The following figure provides an example of this process for a rising threshold trap:



In this example, Trap 1 is generated because it is the first time that the parameter value rises above the Rising Threshold. The next two times the parameter value rises above the Rising Threshold, a trap is not generated because the parameter did not fall below the Falling Threshold. Trap 2 and Trap 3 are generated because the parameter value dropped below the Falling Threshold before exceeding the Rising Threshold.

Enabling or Disabling a Threshold Trap

Each line in the NTREND.INI file and N_NTTREN.INI file contains a parameter that enables or disables the NTREND.NLM software to send traps as determined by the rising and falling thresholds. This parameter is set to 1 to enable the software to send a trap for the values given, or to 0 to disable the software from sending a trap for this parameter.

Controlling Alarm Generation

Each managed server has files that specify which system events result in a trap. On NetWare, the NWTRAP.CFG and NDSTRAP.CFG files are stored in the SYS:\ETC directory. On Windows NT, this file is NTTRAP.INI, which is stored in the MW\INI directory.

On NetWare, the trap configuration file is read only when NWTRAP.NLM is loaded; therefore, any changes made to the file do not take effect until the next time you load NWTRAP.NLM or NDSTRAP.NLM.

IMPORTANT: On a NetWare 3.x server, EDIT.NLM does not have a large enough buffer to edit the NWTRAP.CFG file. To edit the NWTRAP.CFG file, map a drive to the server's SYS: volume and proceed from there.

The .CFG files on NetWare contain the list of supported traps. You can modify the .CFG files or NTTRAP.INI file with the following:

- ◆ Types of alarms forwarded to ConsoleOne
- ◆ Community strings used for sending SNMP traps
- ◆ List of traps to be disabled, using the mask keyword
- ◆ Specific alarms that you want to prevent from forwarding

The configuration file consists of keywords and their associated data (case is ignored). Each keyword must be on a line by itself (except for mask values, where they might span several lines), and must be followed by one or more lines of associated data.

You can place comments anywhere in the file, even between a keyword and its associated information. A comment starts with a number sign (#), and continues to the end of the line.

The following is an example of an NWTRAP.CFG file:

```
#
#####
#NWTRAP.CFG
#
#NWTRAP Configuration File
#
#This file specifies information to be used by NWTRAP.NLM
#The file is read and the parameters set when NWTRAP is loaded. It must
#reside on volume SYS: in the directory SYS:\ETC and must be named
#NWTRAP.CFG to be found by NWTRAP. To change the parameters, first
#edit this file, then unload NWTRAP and load it again. Any changes to this
#file will not take effect until NWTRAP is next loaded. The parameters
#are specified by using a parameter keyword followed by the desired
#parameter value.
#
#####

Community
    Public
Time Interval
    10
Severity
    Warning

mask
# "Memory: Short term alloc failed"
# 1

# "FileSys: Directory write error (no vol)"
# 2

# "FileSys: File write err, by server (no path)"
# 3

# "FileSys: File write err, by user (no path)"
# 4
```

The following sections contain information to help you control alarm generation:

- ◆ “Setting the Time Interval (Management Agent for NetWare Only)” on page 135
- ◆ “Configuring Alarm Severity Levels” on page 136

Setting the Time Interval (Management Agent for NetWare Only)

Sometimes an alarm repeats rapidly (several times per second or per minute) with identical or nearly identical parameters. When this occurs, the second and later alarms within a time interval are usually not as interesting as the first alarm.

To prevent the network and ConsoleOne from being inundated with identical alarms, you can specify a time interval to be applied to every alarm generated. During this interval, alarms that are identical to an initial alarm are discarded.

You can define the time interval in the configuration file as follows:

```
Time Interval
```

```
n
```

where n can take any value from 0 to 232 to indicate the number of seconds that must elapse before a later alarm is not discarded.

The default time interval is 10 seconds.

Configuring Alarm Severity Levels

Use the severity keyword to set a minimum alarm severity level so that traps for lesser severity alarms are not sent.

The severity levels you can set in the NWTRAP.CFG and NTTRAP.INI files are informational, warning, recoverable, critical, and fatal. The following table lists the NetWare severity level and corresponding SNMP and ZfS severity levels.

NetWare Severity Level	SNMP Severity Level	ZfS Severity Level
0 - Informational	Informational	Informational
1 - Warning	Minor	Minor
2 - Recoverable	Major	Major
3 - Critical	Critical	Severe
4 - Fatal	Fatal	Severe
5 - Operation Aborted	Fatal	Severe
Unrecoverable	Fatal	Severe

The default keyword is warning. Under the default, all alarms with a severity level of warning or greater are forwarded.

Defining Recipients for SNMP Alarms

You can configure the Management Agent for NetWare to send SNMP traps (alarms) to the ZfS management server or to other management nodes.

NOTE: For setting trap destinations on Windows NT servers, see the documentation on the SNMP Service provided with the Microsoft Windows NT operating system software.

Steps for designating trap target destinations are described in the following section.

Editing the TRAPTARG.CFG File Manually (Management Agent for NetWare Only)

You can configure trap recipients by manually adding them to the TRAPTARG.CFG file. This is useful for sending traps to third-party management consoles other than the ZfS management server.

You must add trap recipients manually by specifying their addresses in the TRAPTARG.CFG file, which is located in the SYS:\ETC directory of all NetWare servers.

The TRAPTARG.CFG file defines the recipients of SNMP traps. You can use this file to define recipients of SNMP traps over IPX and UDP/IP. The file is fully annotated to show you how to divide the file into IPX and UDP/IP sections and how to write the IPX and IP addresses of recipients.

The TRAPTARG.CFG file is read only when SNMP is loaded. In most cases, this means bringing the server down and restarting it because a variety of modules must be unloaded and reloaded as well. Thus, any changes made to the TRAPTARG.CFG file do not take effect until the next time you load NWTRAP.NLM.

IMPORTANT: The NWALARM.MIB file imports symbols from the Host Resources MIB (RFC1514.MIB), which can also be found in SYS:ZENWORKS\MMS\MWSERVER\MIBCSERVER\MIBSERVERPOOL\MIBPOOL.

Managing Servers

With the Management Agent for NetWare and Management Agent for Windows NT software installed on your NetWare and Windows NT servers, respectively, you can begin collecting data, receive alarm notifications, remotely manage configuration, and generate reports for managed servers.

Server Management tasks you can perform with ZfS include:

- ◆ “Displaying Server Configuration Information” on page 137
- ◆ “Displaying Summary Data” on page 138
- ◆ “Viewing Trend Data” on page 139
- ◆ “Managing Trend Samplings” on page 141
- ◆ “Configuring Server Parameters” on page 142
- ◆ “Executing Server Commands” on page 143

Displaying Server Configuration Information

Server configuration data is organized in a hierarchical listing expanding down from the server object. You can view information about the server's configuration, memory usage, adapters, network interfaces, disks and disk controllers, volumes, queues, users, connections, open files, NLM files (NetWare), and installed software.

To display server configuration information:

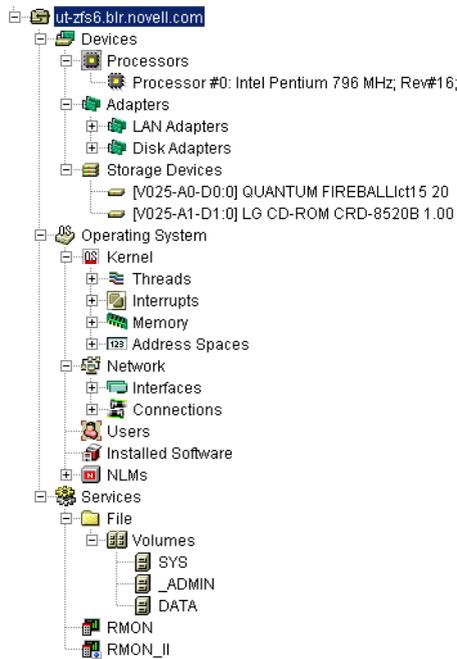
- 1** Locate the server object you want to expand.
- 2** Click the plus sign (+) next to the server object.

The server object opens in the left pane under its parent object and the server contents are displayed. Server data is grouped into the following three categories:

- ◆  Devices
- ◆  Operating System
- ◆  Services

If you are unable to view the above three categories, you must perform probe manageability on the server object. Right-click the server object > select Probe Manageability. The three categories will now be displayed.

- 3** You can drill down into the server configuration farther by clicking the plus signs next to the Devices, Operating System, and Services objects as in the following example.



Displaying Summary Data

The Summary View contains tables of statistics obtained by SNMP GET requests to the Management Agent for NetWare and Management Agent for Windows NT software hosted on managed servers. Statistics are updated dynamically as the server is continually polled for data. Polling utilizes SNMP GET and GET NEXT requests to update the data. You can also control the polling of a selected object by using the stop and refresh functions.

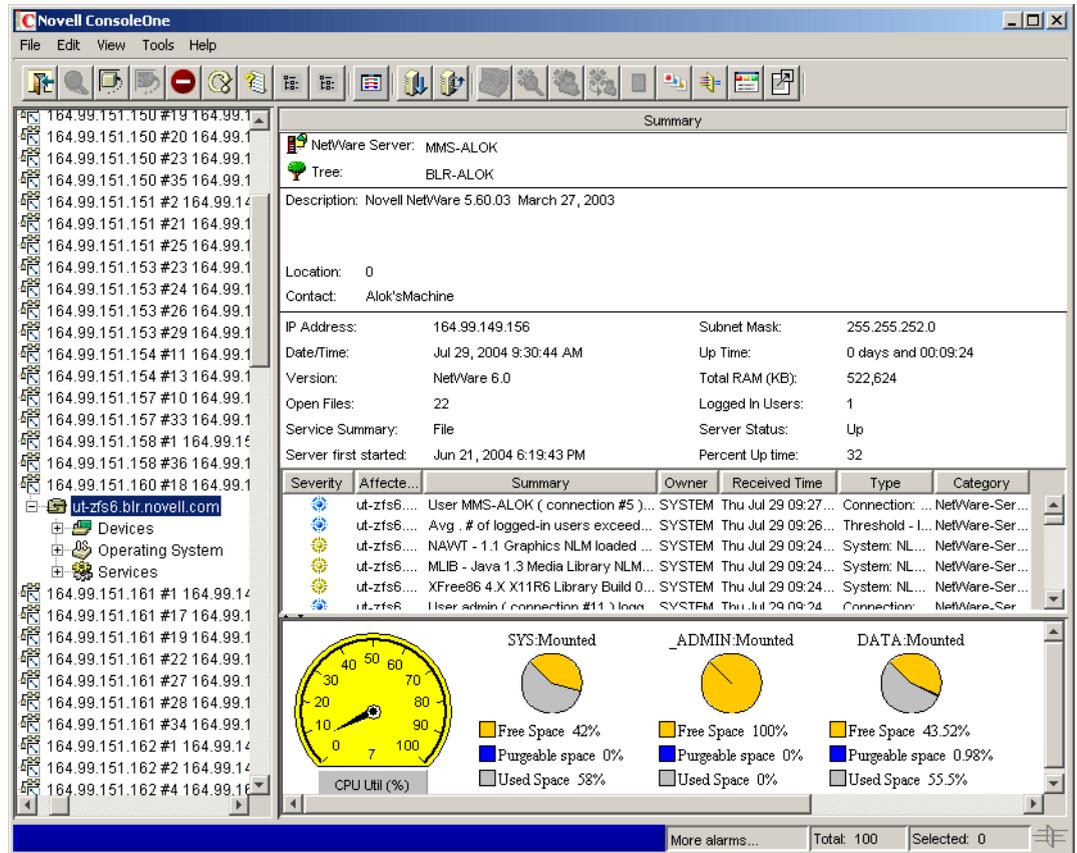
You can view summary data for server, processors, LAN adapters, disk adapters, storage devices, threads, interrupts, memory, address spaces, interfaces, connections, users, installed software, NLM files (NetWare), and volumes. For detailed information about a specific Summary view, see [“Object Hierarchy and View Details” on page 144](#).

To display summary information:

- 1** Right-click the object for which you want to view summary data > click Views > click Summary.

The Summary View is displayed. The following screen shows the Summary View for a server object. The server summary provides descriptive information including the server’s

eDirectory name and tree, IP address, RAM, operating system and version, IPX address, subnetwork mask, up time, logged-in users, open files, and status. In addition, the server summary lists all alarms, affected objects, summary, and owner and volume disk space, trend graphs for cache hits and cache buffers. The Summary view displays graphical indicators of the CPU utilization that depicts the average percentage of time that the CPU was not idle for the past minute.



Viewing Trend Data

On a server managed by Management Agent for NetWare or Management Agent for Windows NT, the agents automatically gather trend data on CPU usage, memory usage, and network interface traffic. You can then view current trend data, or historical trend data by hour, day, week, month, or year from ConsoleOne. In this view, the time interval that is being sampled is displayed on the x-axis. The parameter value over the sample period is plotted on the y-axis. Note that the values on the y-axis use the standard abbreviations K (for kilo), M (for mega), and G (for giga). Therefore, a value of 1K would equal 1000; similarly, a value of 1M would equal 1,000,000.

Monitoring trend data helps you with tasks such as setting trend alarm thresholds, determining who is using the server and when the server is used heavily, troubleshooting problems, balancing loads across multiple servers, and planning resources. You can also export trend view data to popular spreadsheet formats for sharing data with others.

You can view trend data for processors, LAN adapters, storage devices, memory, connections, users, and volumes. For information about a specific trend view, see [“Object View Details” on page 146](#).

To view trend statistics:

- 1 Right-click the object for which you want to view trend data > click Views > click Trend.

The Trend View is displayed. The following sections describe the tasks you can perform using the Trend View:

- ◆ [“Displaying the Legend” on page 140](#)
- ◆ [“Modifying the Time Span” on page 140](#)
- ◆ [“Customizing the Trend View Display” on page 140](#)
- ◆ [“Modifying the Trend View Profile” on page 141](#)

Displaying the Legend

The Trend View legend indicates what each color in the graph represents.

To display the legend:

- 1 Click the Legend button  in the Trend View toolbar.

Modifying the Time Span

The Trend View time span specifies what time period the trend graphs represent. By default, a one-hour history is displayed.

To modify the time span:

- 1 Select a time span from the drop-down list in the Trend View toolbar. You can select from the following time spans:
 - ◆ 1 Hour
 - ◆ 1 Day
 - ◆ 1 Week
 - ◆ 1 Month
 - ◆ 1 Year

Customizing the Trend View Display

The Trend View provides several options for customizing the look of the screen. In customizing the view, you can choose from the following options:

- ◆ [“Displaying Grid Lines” on page 140.](#)
- ◆ [“Stacking and Unstacking Graphs” on page 140.](#)
- ◆ [“Scaling the Y Axis” on page 141.](#)

Displaying Grid Lines

By default, the trend charts do not include grid lines.

To display horizontal and/or vertical grid lines:

- 1 To display horizontal grid lines, select the Horizontal Grid  button in the Trend View toolbar.
- 2 To display vertical grid lines, select the Vertical Grid  button in the Trend View toolbar.

Note that you remove the horizontal or vertical grid lines by clicking the same buttons.

Stacking and Unstacking Graphs

By default, all trends are displayed on a single graph with one vertical axis. However, you can customize the view so that each trend is displayed in its own separate graph.

To stack and unstack graphs:

- 1 To display the trends on separate graphs, click the Strip Chart  button on the Trend View toolbar.
- 2 To display trends on the same graph, click the Stack Chart  button on the Trend View toolbar.

Scaling the Y Axis

To display more useful information on your trend graphs, you may find that you need to modify the scale on the Y axis as follows:

- 1 To increase the scale on the Y axis, click the Increase Y Axis  button, which is located to the left of the graph(s).
- 2 To decrease the scale on the Y axis, click the Decrease Y Axis  button, which is located to the left of the graph(s).
- 3 To scale the Y axis to fit in the window, click the Scale to Fit  button on the Trend View toolbar.

Modifying the Trend View Profile

The Trend View profile represents the set of parameters that are displayed graphically when the Trend View is invoked. You can modify which parameters are displayed in the Trend View by editing the profile.

To edit the profile:

- 1 Click the Profile button  in the Trend View toolbar.
The Profile dialog box is displayed. The parameters that are currently displayed in the Trend View for the object are selected.
- 2 Edit the profile by clicking a parameter name to select or deselect it.
You can Shift+click multiple, consecutive parameters and Ctrl+click multiple, non-consecutive parameters.
- 3 Click OK.

Managing Trend Samplings

You can customize the parameters of the trend data displayed using the following options:

- ♦ [“Modifying Trend Sampling and Intervals” on page 141.](#)
- ♦ [“Modifying Threshold Alarm Settings” on page 142.](#)

Modifying Trend Sampling and Intervals

For each trend for which the server agents collect data, you can set sampling intervals and the number of samples stored on the server as follows:

- 1** Right-click the object > click Properties.
- 2** Click the Trend tab.
- 3** Select the trend parameter you want to modify > click Edit.

The Edit Trend dialog box is displayed. The trend sampling and interval settings are displayed in the Sampling Parameters section of the screen.

- 4** To enable or disable the sampling parameter, select the appropriate value from the State drop-down list.
- 5** To modify the time interval (Sample Interval) at which the trend parameter is sampled, select a value from the Frequency drop-down list.

You can select one of 12 possible time intervals from five seconds to one day.

- 6** Specify the duration of time for which to collect samples by entering a value in the Number of Samples field.

You determine the duration of time for which a parameter is collected by the number of samples (trend buckets) you specify. You must specify a trend bucket for each sample that is collected over a specific period of time. For more information on setting the number of samples required, see [“Setting the Trend Buckets” on page 132](#).

- 7** When you are done modifying the alarm threshold settings, click OK.

Modifying Threshold Alarm Settings

You can set an alarm threshold for each trend parameter for which the Management Agent for NetWare and Windows, collects data. After you set the alarm threshold, the Management Agent for NetWare sends an alarm to ConsoleOne if the trend crosses the threshold you set.

The Management Agent for NetWare tracks both rising and falling alarm thresholds. Each trend parameter has either a rising or a falling threshold associated with it; the type of threshold cannot be changed.

To change alarm thresholds through ConsoleOne:

- 1** Right-click the object > click Properties.
- 2** Click the Trend tab.
- 3** Select the trend parameter for which you want to modify threshold settings > click Edit.

The Edit Trend dialog box is displayed. The threshold alarm settings are displayed in the Rising Alarm Parameters section of the screen.

- 4** To enable or disable the alarm parameter, select the appropriate value from the State drop-down list.
- 5** To set or modify the rising threshold, enter a value in the Rising Threshold field.
- 6** To set or modify the falling threshold, enter a value in the Falling Threshold field.
- 7** When you are done modifying the alarm threshold settings, click OK.

Configuring Server Parameters

In order to correct an alarm condition, fine-tune server performance, or fix other problems detected on a server, you need to modify the server configuration. Server configuration can be adjusted from ConsoleOne on any NetWare server hosting the Management Agent for NetWare. SET

parameters, usually set at the server console or through a remote console, can be configured from ConsoleOne interface. From ConsoleOne, you can see the current settings, change one or more settings, and confirm your settings before adjustments are sent to the server.

For parameter values and descriptions, see the NetWare server documentation. This information is generally found in the Utilities Reference document.

To view or modify the NetWare SET parameters from ConsoleOne:

- 1** Drill down into the server you want to configure by clicking the plus sign (+) next to the server object.
- 2** Right-click the Operating System object > click Properties.
The Set Parameters tab is displayed. This tab page lists the NetWare SET parameters and their current values.
- 3** Click the down-arrow icon on the Set Parameters tab > click the category of SET parameters you want to display.
You can choose from the following categories: Communications, Directory Caching, Directory Services, Disk, Error Handling, File Caching, File System, Licensing Services, Locks, Memory, Miscellaneous, Multiprocessor, NCP, Service Location Protocol, Time, or Transaction Tracking.
- 4** Select the parameter you want to modify > click Edit.
The Edit Parameters dialog box is displayed.
- 5** Enter the new parameter value in the appropriate field.
- 6** Indicate when you want the parameter change to take effect by selecting the appropriate radio button from the Apply Value box. You can choose to apply the change at the following times:
 - ◆ Now, until reboot
 - ◆ Only after reboot
 - ◆ Now, and after reboot
- 7** Click OK.

Executing Server Commands

You can execute the following frequently used NetWare server commands from ConsoleOne.

- ◆ [“Loading and Unloading an NLM” on page 143](#)
- ◆ [“Mounting and Dismounting Volumes” on page 143](#)
- ◆ [“Clearing a Server Connection” on page 144](#)
- ◆ [“Restarting a Server” on page 144](#)
- ◆ [“Shutting Down a Server” on page 144](#)

Loading and Unloading an NLM

To load or unload an NLM from ConsoleOne:

- 1** Right-click the NLM object > select a command from the menu as follows:
 - ◆ To load the NLM, select Load nlm
 - ◆ To unload the NLM, select Unload nlm

Mounting and Dismounting Volumes

To mount or dismount a volume:

- 1 Right-click the volume object > click Mount Volume.
or
Right-click the volume object > click Dismount Volume.
The system displays a confirmation box.
- 2 Click OK.

Clearing a Server Connection

You can clear a server connection when the server has crashed and left open files on the server or before bringing down the server. This is equivalent to the CLEAR STATION command that you can execute from the server console.

To clear a server connection from ConsoleOne:

- 1 Locate the connection you want to close by expanding the following objects: Server > Operating System > Network > Connections.
- 2 Right-click the connection you want to close > click Clear Connection.

Restarting a Server

To restart a server from ConsoleOne:

- 1 Right-click the server object > click Restart Server.

Shutting Down a Server

To shut down a server from ConsoleOne:

- 1 Right-click the server object > click Down Server.

Object Hierarchy and View Details

When you expand a managed server object, you can view details about the contents of the server. The following sections detail the available objects on a managed server and provide information about the statistical information available in the views for each object. This topic contains the following sections:

- ◆ [“Object Hierarchy” on page 144](#)
- ◆ [“Object View Details” on page 146](#)

Object Hierarchy

The following table shows the hierarchy of available objects on a managed server along with their associated icons. For more information about the available views associated with an object, follow the corresponding link.

Category Container	Sub-category Containers	Object Containers	Objects	
 Devices	 "Processors" on page 146		 Processor	
	 "Printers" on page 163		 Printers	
	 "Adapters" on page 149	 LAN Adapters	 Adapter	
		 Disk Adapters	 Adapter	
	 "Storage Devices" on page 148		 Storage Device	
	 "Other Devices" on page 163		 Keyboard	
			 Mouse	
 "Ports" on page 164		 Parallel Port		
		 Serial Port		
 Operating System	 Kernel	 "Threads" on page 150	 Thread	
		 "Interrupts" on page 151	 Interrupt	
		 "Memory" on page 152	 Memory	
		 "Address Spaces" on page 154	 Address Space	
		 "Network" on page 155	 "Interfaces" on page 155	 Interface
			 "Connections" on page 157	 Connection
		 "Users" on page 158		 User
	 "Installed Software" on page 159		 Software	
 "NLM" on page 159		 NLM		

Category Container	Sub-category Containers	Object Containers	Objects
 Services	 File	 "Volumes" on page 160	 Volume
	 Print	 "Queues" on page 162	 Queue

Object View Details

The following sections provide details about the statistical information available in each object view:

- ◆ ["Processors" on page 146](#)
- ◆ ["Storage Devices" on page 148](#)
- ◆ ["Adapters" on page 149](#)
- ◆ ["Threads" on page 150](#)
- ◆ ["Interrupts" on page 151](#)
- ◆ ["Memory" on page 152](#)
- ◆ ["Address Spaces" on page 154](#)
- ◆ ["Network" on page 155](#)
- ◆ ["Interfaces" on page 155](#)
- ◆ ["Connections" on page 157](#)
- ◆ ["Users" on page 158](#)
- ◆ ["Installed Software" on page 159](#)
- ◆ ["NLM" on page 159](#)
- ◆ ["Volumes" on page 160](#)
- ◆ ["Queues" on page 162](#)
- ◆ ["Printers" on page 163](#)
- ◆ ["Other Devices" on page 163](#)
- ◆ ["Ports" on page 164](#)

Processors

Viewing processor speed helps you analyze and balance loads across servers. Viewing processor utilization data helps you detect problems with utilization and determine when server load is light enough to schedule tasks such as server backups. The server operating system (OS) automatically determines the CPU speed and is reported based on the OS data.

Processor speed is a major determinant of server performance. Therefore, it is important to know the processor speed of your servers when analyzing server load and balancing load across multiple servers. For example, one server might be handling twice as many users as another, but if the processor is twice as fast, the load might still be distributed correctly.

You should maintain a baseline of processor utilization for a server so that you can recognize when a server's processor utilization is higher than normal.

You can display the following views of information about the processors on your managed servers:

- ◆ “Processors Summary View” on page 147
- ◆ “Processor Summary View” on page 147
- ◆ “Processors Trend View” on page 147

Processors Summary View

You can access the **Summary View** for the Processors object container after expanding the following server objects: Devices > Processors. This view displays the following information for each processor object in the container:

- ◆ **Processor Number:** A unique number assigned to the processor.
- ◆ **Status:** The status of the processor is either online or offline.

The following statistics are displayed only if the processor is online:

- ◆ **Utilization %:** Processing load on this processor for the last second, expressed as a percentage.
- ◆ **Interrupts Processed:** Number of interrupts fired on this processor in the last second.
- ◆ **Time Spent in Interrupts Last Second, in Microseconds:** The amount of time in microseconds that the processor spent processing interrupts in the last second.
- ◆ **Number of Bound Threads:** The number of threads that have been bound to this processor. Threads that are bound to a processor run only on that processor. Unbound threads can be migrated from one processor to another when required.

Processor Summary View

You can select the **Summary View** for an individual processor after expanding the following server objects: Devices > Processors > *processor #x*. This view displays the following information:

- ◆ **Processor Number and Status:** A unique number assigned to the processor along with its current status. The status can be online or offline.

The following statistics are displayed only if the processor is online.

- ◆ **Utilization %:** The processing load on this processor for the last second, expressed as a percentage.
- ◆ **Interrupts Processed:** The number of interrupts fired on this processor in the last second.
- ◆ **Time Spent in Interrupts Last Second, in Microseconds:** The amount of time in microseconds that the processor spent processing interrupts in the last second.
- ◆ **Number of Bound Threads:** The number of threads that have been bound to this processor. Threads that are bound to a processor run only on that processor. Unbound threads can be migrated from one processor to another when required.

Processors Trend View

You can access the **Trend View** for the Processors object container after expanding the following server objects: Devices > Processors. This view displays the following graph for each processor:

- ◆ **CPU Utilization (avg. %):** The processing load on the processor for the last second, expressed as a percentage. This information is displayed only if the processor is online.

Storage Devices

You can get detailed information about the disk drives in a managed server, including disk size in megabytes, disk types, block size, and so on.

You can also view partition information for each disk drive. Partition information is especially informative because you can determine whether a partition is fault tolerant and whether the hard disk is losing data integrity.

Fault tolerance of a NetWare partition is part of the detailed information provided by Zfs server management. To determine whether a hard disk is losing data integrity, examine the redirected area. A number in the redirected area indicates the number of data blocks that have been redirected to the Hot Fix Redirection Area to maintain data integrity. The higher the redirected area number, the more faulty blocks there are on the hard disk. A redirected area growing over a period of time indicates a hard disk going bad.

On a NetWare server managed by the Management Agent for NetWare or a Windows NT server managed by the Zfs management agent, the Agent automatically gathers trend data on CPU usage, memory usage, and network interface traffic. In Zfs, you can view current trend data, or historical trend data by hour, day, month, or year. Monitoring trend data helps you with tasks such as setting alarm thresholds, determining who is using the server and when the server is used heavily, troubleshooting problems, balancing loads across multiple servers, and planning resources.

You can display the following views of information about the storage devices on your managed servers:

- ◆ “Storage Devices Summary View” on page 148
- ◆ “Storage Device Summary View” on page 149
- ◆ “Storage Devices Trend View” on page 149

Storage Devices Summary View

You can select the **Summary View** for the Storage Devices container object after expanding the following server objects: Devices > Storage Devices. This view provides the following information for each storage device on the server:

- ◆ **Disk Name:** The name of the disk drive.
- ◆ **Size (KB):** The total size of the disk drive in kilobytes.
- ◆ **Access:** Whether the disk drive is readable and writable or just readable.
- ◆ **Status:** Whether the disk drive is operational.
- ◆ **Type:** The type of media. Media types can include hard disk, floppy disk, tape, optical disk (read-only, write once read many, and read/write), or RAM disk. If unidentifiable, other or unknown is listed in this field.
- ◆ **Driver Description:** The name of the driver used by the disk drive.
- ◆ **Block Size:** The number of blocks used on the disk in kilobytes.
- ◆ **Heads:** the number of read/write heads on the disk drive.
- ◆ **Cylinders:** The number of cylinders on the disk drive.
- ◆ **Sectors/Track:** The number of sectors per track on the disk drive.
- ◆ **SCSI Target ID:** The target address for SCSI controllers or the unit number for other devices and the logical unit number for SCSI devices or the number zero for other devices.

Storage Device Summary View

You can display the **Summary View** for an individual storage device by expanding the following server objects: Devices > Storage Devices > *storage_device_x*. This view displays the following information:

- ◆ **Disk Name:** Name of the disk drive.
- ◆ **Logical ID:** The number assigned to a logical partition for identification.
- ◆ **Physical ID:** The number assigned to a physical partition for identification.
- ◆ **Type Partition:** The type of partition, including DOS, NetWare, and UNIX* partitions.
- ◆ **Size (KB):** The size of the partition, in kilobytes.
- ◆ **Redirection Area:** The size of the entire Hot Fix Redirection Area.
- ◆ **Redirected Area:** The number of bad blocks Hot Fix found.
- ◆ **Reserved Area:** The number of Hot Fix redirection blocks reserved for system use.
- ◆ **Fault Tolerance:** The type of fault tolerance used. The possible fault tolerance types are duplex and mirrored. If there is no fault tolerance, this field contains the value None.

Storage Devices Trend View

You can select the Storage Devices **Trend View** after expanding the following server objects: Devices > Storage Devices. This view provides the following information:

- ◆ **File System Reads (#/min):** Depicts the number of file system reads made per minute on multiple or single storage devices.
- ◆ **File System Writes (#/min):** Depicts the number of file system writes made per minute on multiple or single storage devices.
- ◆ **File System Reads (KB/min):** Depicts the number of file system reads per kilobyte volume made on multiple or single storage devices.
- ◆ **File System Writes (KB/min)** Depicts the number of file system writes per kilobyte volume made on multiple or single storage devices.
- ◆ **Free Redirection Area (%):** Depicts the percentage of total volume allocated to the disk redirection area.

Adapters

You can get detailed information about the network and disk adapters in a managed server, including I/O port, memory address, and interrupt configuration.

You can use this data to detect configuration problems such as the same address or interrupt is configured for two boards inside the server, or for a board and a component of the server's hardware. No two boards can use the same I/O port, memory address, and interrupt.

Problems with LAN adapters cause network problems, such as servers and workstations not being able to communicate. You can use the data collected on the LAN adapter to determine whether the frame type used by a network board is bound to a supported protocol. (A single network board might be bound to several protocols.) You can immediately tell whether a problem is due to something as simple as using the wrong frame type on the workstation (for example, an Ethernet_II frame type on the server and the Ethernet_802.2 frame type on the workstation).

You can display the following views of information about the adapters on your managed servers:

- ◆ “Adapters Summary View” on page 150
- ◆ “Adapters Trend View” on page 150

Adapters Summary View

You can select the Adapters **Summary View** after expanding the following server objects: Devices > Adapters > *adapter_x*. This view provides the following information:

- ◆ **Description:** The type of adapter hardware. This field can include the following types of information: manufacturer, model, and version. Or, for network boards, this field may contain a short board name and the board’s burned-in MAC address.
- ◆ **Type:** The type of adapter (for example, network card or disk storage).
- ◆ **Devices Attached:** The number of devices associated with an adapter (for example, the number of drives attached to the disk controller).
- ◆ **Driver Description:** Description of the driver for this adapter.
- ◆ **Version:** The version number of the driver software.
- ◆ **Interrupt Number:** The unique interrupt number used by the adapter.
- ◆ **I/O Port:** The unique I/O port block used by the adapter.
- ◆ **Memory:** The unique memory address space used by the adapter.
- ◆ **DMA:** The Direct Memory Access (DMA) Channel used by the adapter.
- ◆ **Slot:** The slot in which the adapter is installed.

Adapters Trend View

You can select the Adapters **Trend View** after expanding the following server objects: Devices > Adapters > *adapter_x*. This view provides the following graphs:

- ◆ **LSL Packets Received:** Depicts the number of LSL packets received by the adapter.
- ◆ **LSL Packets Transmitted:** Depicts the number of LSL packets transmitted by the adapter.
- ◆ **Packets Received:** Depicts the total number of packets received by the adapter.
- ◆ **Packets Transmitted:** Depicts the total number of packets transmitted by the adapter.

Threads

You can display information for all threads currently running on a managed server. A thread is recognized as an independent unit of execution.

You can display the following view of information about the threads on your managed servers:

- ◆ “Threads Summary View” on page 150

Threads Summary View

You can select the Threads **Summary View** after expanding the following server objects: Operating System > Kernel > Threads. This view provides the following information:

- ◆ **Name:** The application thread name.
- ◆ **Share Group:** The Application share groups and their associated threads and shares.
- ◆ **Parent Module:** Module (NLM) associated with this thread.

- ◆ **State:** The state of the thread, which can be one of the following: initializing, invalid, ready, running, suspended, terminated, or zombie.
- ◆ **Suspended Due To:** Reason the thread is suspended. If the thread is not in a suspended state, this field is blank.
- ◆ **Execution Time, Microseconds:** Amount of time in the last second that the processor spent executing the thread's code.
- ◆ **Stack Size, Bytes:** Size of the thread's stack.
- ◆ **Soft Affinity:** Processor on which the thread preferentially executes, but from which it can migrate when necessary.
- ◆ **Hard Affinity:** Indicates whether the thread is explicitly bound to a specified processor for the thread's lifetime. If the thread runs only on a specified processor, it is able to exploit the processor's cache state. If the thread is allowed to run on any available processor, the field value is zero.

Interrupts

You can display information for the registered interrupts on a managed server. On a multiprocessing system, interrupt information is displayed for all processors combined and individually for each online processor.

You can display the following views of information about the interrupts on your managed servers:

- ◆ [“Interrupts Summary View” on page 151](#)
- ◆ [“Interrupts Service Routines View” on page 152](#)

Interrupts Summary View

You can select the Interrupts [Summary View](#) after expanding the following server objects: Operating System > Kernel > Interrupts. This view provides the following information:

- ◆ **Name:** The name of the interrupt routine.
- ◆ **Interrupt Number:** Number for this service routine.
- ◆ **Processor:** Number of the processor.
- ◆ **Type:** The type of interrupt service routine. It can be one of the following:
 - ◆ **Bus:** A device I/O interrupt that is used (for example, by disk or LAN drivers).
 - ◆ **Local:** A hardware platform-specific interrupt local to an individual processor.
 - ◆ **System:** An interrupt category that is reserved for systems with unique interrupt requirements.
 - ◆ **Interprocessor:** An interrupt that is generated by one processor to affect another processor.
 - ◆ **Timer:** An interrupt that provides timer services for the OS as well as preemption support. (In multiprocessing systems, timer interrupts are local to a processor.)
- ◆ **Service Routines:** Number of service routines that are launched when this interrupt occurs.
- ◆ **Interrupt Occurrences:** Number of times in the last second that the interrupt occurred and was processed.
- ◆ **Execution Time:** Amount of time in the last second that the processor spent processing this interrupt.

- ◆ **Spurious Interrupts:** Number of times since the server started that an interrupt fired that should not have occurred.

Interrupts Service Routines View

The Interrupts Service Routines View provides information about the memory address spaces defined on the server.

NetWare runs in the OS address space (kernel), along with LAN drivers, storage device drivers, MONITOR, and network management agents (NMAs). OS address space is backed by physical memory.

All other address spaces are user space (ring 3) and are backed by virtual memory. Applications running in user space cannot cause the server to abend if the address space faults.

You can select the Service Routines View after expanding the following server objects: Operating System > Kernel > Interrupts. This view provides the following information:

- ◆ **Name:** The name of the interrupt service routine.
- ◆ **Service Routine Number:** Service Routine Number associated with this service routine.
- ◆ **Processor Number:** Processor number this routine is running on.
- ◆ **Interrupt Number:** Interrupt number associated with this service routine.
- ◆ **Interrupts Processed Last Second:** Number of interrupts that were processed by the ISR during the last second.

Memory

You can display the following views of information about the memory on your managed servers:

- ◆ [“Memory Summary View” on page 152](#)
- ◆ [“Memory Trend View” on page 152](#)
- ◆ [“Disk Cache View” on page 153](#)
- ◆ [“Virtual Memory View” on page 154](#)

Memory Summary View

You can select the Memory [Summary View](#) after expanding the following server objects: Operating System > Kernel > Memory. This view provides the following information:

- ◆ **Type:** The type of memory (for example, DOS, allocated memory, cache buffers, or code and data memory).
- ◆ **Unit Size (bytes):** The size of the memory allocation.
- ◆ **Total (KB):** The number of memory units × the unit size.
- ◆ **Units Used:** The number of memory units that have been allocated.
- ◆ **Used (KB):** The number of KB of memory that has been allocated.

The Memory Summary View also provides a pie chart depicting memory usage on the system.

Memory Trend View

You can select the Memory [Trend View](#) after expanding the following server objects: Operating System > Kernel > Memory. This view provides the following graphs:

- ♦ **Cache Buffers (%):** The percentage of memory allocated to cache buffers.
- ♦ **Code and Data Memory (%):** The percentage of memory allocated to code and data.
- ♦ **Allocated memory (%):** The amount of allocated memory.
- ♦ **Dirty Cache Buffers (%):** The amount of dirty cache buffer memory.

Disk Cache View

This view displays utilization for disk cache memory. Use cache utilization statistics to determine when you need to install more RAM for cache. You can select this view after expanding the following server objects: Operating System > Kernel > Memory. It provides the following information:

- ♦ **Short Term Cache Hits %:** Percentage of requests in the last second for disk blocks that were already in cache memory. When the requested data is already in memory, disk reads don't need to be made. If this value falls below 98%, consider installing more RAM for cache. Also compare with Long Term Cache Hits.
- ♦ **Short Term Cache Dirty Hits %:** Percentage of requests in the last second for disk blocks that were already in cache memory but were dirty. Dirty cache must be written to disk before being used. Also check Long Term Dirty Cache Hits and LRU Sitting Time.
- ♦ **Long Term Cache Hits %:** Cumulative percentage of requests for disk blocks that were already in cache. When the requested data is already in memory, disk reads don't need to be made. Use this cumulative percentage to assess overall disk cache utilization. If this value falls below 90%, install more RAM for cache.
- ♦ **Long Term Cache Dirty Hits %:** Cumulative percentage of requests for disk blocks that were already in cache memory but were dirty. (Before dirty cache can be used, it must be written to disk.) Use this cumulative percentage to assess overall disk cache utilization. If this value is high or steadily incrementing, add more RAM for cache. Also check LRU Sitting Time.
- ♦ **Total Cache Blocks Allocated:** Cumulative number of requests for disk cache blocks that have been made since the server was started or rebooted. This value is the sum of the values of Allocated from Available List and Allocated from Least Recently Used (LRU). If the value of Allocated from Available is much higher, the server has sufficient RAM for cache. If the value of Allocated from LRU is high, install more RAM for cache.
- ♦ **Cache Blocks Allocated from Available List:** Number of requests for disk cache blocks that were filled by blocks in the available list (blocks that were not being used). When there are no free blocks available, requests are filled from the LRU list of cache blocks. If this value is much higher than the Allocated from LRU value, the server has sufficient RAM for cache.
- ♦ **Cache Blocks Allocated from LRU:** Number of requests for disk cache blocks that were filled by blocks from the Least Recently Used cache blocks. The system writes pending requests from the LRU cache block to disk then frees the block for the current request. Because LRU caches used only when no other cache is available, a steadily incrementing count indicates more RAM is needed.
- ♦ **Number of Times in Last 10 Minutes that the OS Had to Wait:** Number of times in the last 10 minutes that the OS waited for an LRU block in order to fulfill a request. If this value is greater than 7, install more RAM for cache.
- ♦ **Number of Times OS Had to Wait:** Number of times that the OS waited for an LRU block in order to fulfill a request.

- ◆ **Total Number of Times the Write Request Was Delayed:** Number of times a write request was delayed because there were too many writes to perform or because the disk channel was busy. A high value indicates either that the disk channel has too much I/O traffic or that you need to install more RAM for cache.
- ◆ **Number of Times the Request Was Re-tried:** Number of times a disk cache request had to be retried because the target block was being used. If this value is high or steadily incrementing, install more RAM for cache.

Virtual Memory View

This view displays information about the virtual memory system. Use these statistics to monitor the efficiency of server memory usage. If these values are fairly stable over time and if server performance is satisfactory, the server has adequate memory for its load. For example, if the value of Page faults increases, this indicates that the server performance is degrading. Conversely, if the Free swap pages value increases, it is an indication of better server performance.

You can select this view after expanding the following server objects: Operating System > Kernel > Memory. It provides the following information:

- ◆ **Total Page-In Requests:** Number of requests that were made to move virtual memory from swap files since the server was started (server up time).
- ◆ **Page-In Requests in Last 5 Seconds:** Number of requests to move 4 KB virtual memory pages from swap files.
- ◆ **Total Page-Out Requests:** Number of requests that were made to move virtual memory to swap files since the server was started (server up time).
- ◆ **Page-Out Requests in Last 5 Seconds:** Number of requests to move 4 KB virtual memory pages to swap files.
- ◆ **Total Swap Pages:** Number of 4 KB pages in this server's virtual memory system. (The size of the swap file in memory pages is the total number of bytes divided by 4 KB.) The size of the swap file grows or shrinks dynamically to match the memory requirements of the server's load.
- ◆ **Free Swap Pages:** Number of 4 KB pages that are available for use by the virtual memory system.
- ◆ **Reserved Swap Pages:** Number of 4 KB pages that are reserved by the virtual memory system.
- ◆ **Total Page Faults:** Number of times the virtual memory system retrieved from the swap file since the server was started (server up time).
- ◆ **Page Faults in Last 5 Seconds:** Number of times in the last five seconds that the virtual memory system retrieved from the swap file. (This means that accessed memory wasn't backed by physical memory.)

Address Spaces

NetWare runs in the OS address space (kernel) along with LAN drivers, storage device drives, MONITOR, and network management agents (NMAs). OS address space is backed by physical memory.

All other address spaces are user space (ring 3) and are backed by virtual memory. Applications running in user space cannot cause the server to abend if the address space faults.

You can display the following view of information about address spaces on your managed servers:

- ◆ [“Address Spaces Summary View” on page 155](#)

Address Spaces Summary View

You can select the Address Spaces [Summary View](#) after expanding the following server objects: Operating System > Kernel > Address Spaces. This view provides the following information:

- ◆ **Name:** Name of the virtual memory address space where this module runs.
- ◆ **Number of NLMs Loaded:** Count of NLM programs loaded in this address space. NetWare, LAN drivers, storage device drivers, MONITOR, and Network Management Agents (NMAs) are loaded in OS address space (kernel). A server application, such as GroupWise®, Lotus Notes*, or an Oracle* database, can be loaded in its own address space (user space or ring 3).
- ◆ **Mapped Pages:** Total number of physical memory pages backing this address space. Note that the OS address space (kernel) is the only address space backed by physical memory.
- ◆ **Restarted:** Total number of times this address space faulted and restarted automatically. A value of zero (0) indicates that no fault has occurred. A non-zero value indicates that an address space has faulted and recovered. Follow online Troubleshooting documentation for core dump instructions for address spaces.
- ◆ **Memory in Use, Bytes:** Amount of allocated memory in use.
- ◆ **Memory Not in Use, Bytes:** Amount of unused allocated memory.
- ◆ **Memory As Overhead, Bytes:** Amount of memory used for managing the allocation pool plus the amount of memory fragmentation.
- ◆ **Total Blocks:** Number of memory blocks that are in use and that are available at the request of the NLM.
- ◆ **Blocks in Use:** Number of memory blocks that were allocated and used.
- ◆ **Block Not Used:** Number of memory blocks that were allocated but not used.

Network

You can display the following view of information about the network activity on your managed server:

- ◆ [“Network Trend View” on page 155](#)

Network Trend View

You can access the [Trend View](#) for the Network object container after expanding the following server objects: Operating System > Network. This view displays the following graph for each network adapter:

- ◆ **Packets Received (KB/min):** The number of kilobytes received by the adapter for the last minute.

Interfaces

You can display the following view of information about the network interfaces on your managed server:

- ◆ [“Interfaces Summary View” on page 156](#)
- ◆ [“Interfaces Statistics View” on page 156](#)

Interfaces Summary View

You can access the **Summary View** for the Network object container after expanding the following server objects: Operating System > Network > Interfaces.

This view displays the following information:

- ◆ **Frame Type:** The frame type that is bound to this logical board.
- ◆ **MAC Address:** The MAC address of the interface.
- ◆ **Description:** Text describing the interface board.
- ◆ **Line Speed:** The number of bits per second transmitted on this board.
- ◆ **Type:** The type of interface (for example, Ethernet CSMACD).
- ◆ **Logical Board #:** The number assigned to this logical board.
- ◆ **Logical Board Name:** The name assigned to this logical board.
- ◆ **Protocols:** The protocols to which the logical board is bound (for example, IP, ARP, or IPX).

Interfaces Statistics View

You can access the Statistics View for the Network object container after expanding the following server objects: Operating System > Network > Interfaces.

This view displays the following information:

- ◆ **Frame Type:** The frame type that is bound to this logical board.
- ◆ **MAC Address:** The MAC address of the interface.
- ◆ **MTU:** The size of the largest datagram which can be sent/received on the interface.
- ◆ **Admin Status:** The desired state of the interface.
- ◆ **Oper Status:** The current operational state of the interface.
- ◆ **Bytes In:** The total number of bytes received on the interface.
- ◆ **Bytes Out:** The total number of octets transmitted out of the interface.
- ◆ **Ucast Packets In:** The number of subnetwork-unicast packets delivered to a higher-layer protocol.
- ◆ **Ucast Packets Out:** The total number of packets that higher-level protocols requested be transmitted to a subnetwork-unicast address.
- ◆ **Nucast Packets In:** The number of non-unicast packets delivered to a higher-layer protocol.
- ◆ **Nucast Packets Out:** The total number of packets that higher-level protocols requested be transmitted to a non-unicast address.
- ◆ **Discards In:** The number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being deliverable to a higher-layer protocol.
- ◆ **Discards Out:** The number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted.
- ◆ **Errors In:** The number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol.
- ◆ **Errors Out:** The number of outbound packets that could not be transmitted because of errors.

- ◆ **Unknown Protocols In:** The number of packets received via the interface which were discarded because of an unknown or unsupported protocol.

The Clear Counters button in this view resets the values only on the Management Console and not on the Server. This is done to enable the user to get the current data from the server.

Connections

You can display the following views of information about the connections on your managed server:

- ◆ “Connections Summary View” on page 157
- ◆ “Connections Trend View” on page 158
- ◆ “Open Files View” on page 158

Connections Summary View

The Connections **Summary View** displays information and statistics for the connections on the selected server. For example, this view displays the number of files currently being accessed by the server and by other clients. Certain files, such as hidden files that support eDirectory, are always open. You can select this view after expanding the following server objects: Operating System > Network > Connections > *connection_x*.

This view provides the following information:

- ◆ **Connection# Login Name:** A string indicating the connection number and login name. Note that connection 0 (zero) is used by the system. The login name is the eDirectory full distinguished name where applicable.
- ◆ **Client Address:**
 - IP: *xxx.xxx.xxx.xxx:port number*
 - IPX: *network:node:socket*
- ◆ **Connection Time:** The date and time the connection was established.
- ◆ **Privileges:** A connection can have one or more of the following privileges:
 - ◆ Supervisor
 - ◆ Operator
 - ◆ Auditor
 - ◆ High_Privilege
 - ◆ Second_Authentication
 - ◆ Second_High_Privilege
- ◆ **Status:** The status can be one of the following:
 - ◆ Not logged in
 - ◆ Logged in
 - ◆ Need security change
 - ◆ MacStation
 - ◆ Connection abort
 - ◆ Audited

- ◆ Authenticated temporary
- ◆ Audit connection recorded
- ◆ DS audit connection recorded
- ◆ Logout in progress
- ◆ **Read (bytes):** Number of bytes the connection has read since it was established.
- ◆ **Written (bytes):** Number of bytes the connection has written since it was established.
- ◆ **NCP Requests:** Number of NCP requests the connection has made since it was established.
- ◆ **Open Files:** Number of files that are currently opened by the connection.
- ◆ **Locked Records:** Number of file records that are currently locked by the connection.

Connections Trend View

You can select the Connections **Trend View** after expanding the following server objects: Operating System > Network > Connections > *connection_x*. This view provides the following graphs:

- ◆ **Connections (avg. #):** The average number of connections over the last sample interval.

Open Files View

The Connection Open Files View displays information and statistics for the connection on the server. For example, this view displays the number of files currently being accessed by the server and by other clients. Certain files, such as hidden files that support eDirectory, are always open. You can select this view after expanding the following server objects: Operating System > Network > Connections > *connection_x*. This view provides the following information:

- ◆ **Filename:** The name of the open file, including the directory path.
- ◆ **Login Name:** The name of the user (if any) who opened the file. If the file was opened by the system or by an NLM, the Login Name will be a zero-length string.
- ◆ **Volume Name:** The physical name of the NetWare volume containing the open file.
- ◆ **Directory Number:** A number that uniquely identifies an open file within a NetWare volume.
- ◆ **Volume ID:** A number that uniquely identifies a NetWare volume. The value of this object for a particular volume has the same value as the nwVolID object for the same volume.

Users

You can display the following views of information about the users on a selected server:

- ◆ **“Users Summary View” on page 158**
- ◆ **“Users Trend View” on page 159**

Users Summary View

The Users **Summary View** provides information about the users who access the selected server. You can select this view after expanding the following server objects: Operating System > Users. This view provides the following information about each user:

- ◆ **Login Name:** The login name of the user.
- ◆ **Disk Usage:** The amount of disk space the user has used.

- ◆ **Last Login:** The date the user last logged in to the server.
- ◆ **Account Status:** Indicates whether the user account is valid.
- ◆ **Password:** Indicates whether the user's password is valid.
- ◆ **Real Name:** The user's eDirectory real name.
- ◆ **Bad Login:** The number of failed login attempts for the user. The number 65535 displayed in this view indicates that you have exhausted the maximum number of attempts to login.
- ◆ **Bad Login Address:** The network address of the location from which the user login failed, if any.

Users Trend View

The Users **Trend View** provides information about the users who access the selected server. You can select this view after expanding the following server objects: Operating System > Users. This view provides the following graph:

- ◆ **Logged-In Users (avg. #):** Depicts the average number of users logged in to the server.

Installed Software

You can display the following view of information about the software that is installed on a selected server:

- ◆ [“Installed Software Summary View” on page 159](#)

Installed Software Summary View

The Installed Software **Summary View** provides information about the software installed on the selected server. You can select this view after expanding the following server objects: Operating System > Installed Software. This view provides the following information:

- ◆ **Name:** The name of the installed software module.
- ◆ **Type:** The type of software (for example, device drivers, applications, or operating system).
- ◆ **Date Installed:** The date the software was installed.

NLM

You can display the following views of information about the NLM software on a managed server:

- ◆ [“NLM Summary View” on page 159](#)
- ◆ [“Resource Tag View” on page 160](#)

NLM Summary View

The NLM **Summary View** provides information about a selected NLM. You can select this view after expanding the following server objects: Operating System > NLMs > *nlm_x*. This view provides the following information:

- ◆ **Name:** The name of the NLM.
- ◆ **Version:** The version number of the NLM.
- ◆ **Released:** The date and time the NLM was released.

- ◆ **Memory (bytes):** The total memory in bytes used by this NLM. This is a composite total of short term memory, semi-permanent memory, and non-movable memory, cache memory allocated by the NLM plus the sizes of the code, and data sections of this instance of an NLM.
- ◆ **Description:** A text string that describes the NLM.
- ◆ **Copyright:** The copyright string for the NLM.

Resource Tag View

You can select the NLM Resource Tag View after expanding the following server objects: Operating System > NLMs > *nlm_x*. This view provides the following information:

- ◆ **Description:** The name that the owning module assigned to this resource tag.
- ◆ **Number in Use:** The number of instances of the resource tag.
- ◆ **Resource Type:** The type of resource tag that is being tracked (for example, semaphores or processors).
- ◆ **Address Space:** Name of the address space where the module that owns the resource tag is running.

Volumes

NetWare server disk storage space is divided into volumes. You can view various data about the volumes mounted on a server, such as size, free space, how the volumes are distributed across disks, and which users are using the space. For individual volumes you can view data on configuration, open files, segments, and usage. The available views of data include:

- ◆ “Volume Summary View” on page 160
- ◆ “Volume Trend View” on page 161
- ◆ “Open Files View” on page 161
- ◆ “Volume Segment View” on page 161
- ◆ “Volume Usage View” on page 162

Volume Summary View

The Volume **Summary View** provides details about a single volume. You can select this view after expanding the following server objects: Services > File > Volumes > *volume_x*. This view provides the following information:

- ◆ **Size (KB):** The size of the volume in kilobytes.
- ◆ **Free (KB):** The amount of free space on the volume in kilobytes. As files are added or expanded, this number approaches zero. A pie chart shows you how much of the total volume size is free.
- ◆ **Used (KB):** The amount of space, which is determined by subtracting the free disk space from the total volume size.
- ◆ **Status:** Whether the volume is mounted. If the volume is not mounted, only the volume name is listed.
- ◆ **Namespaces:** Namespaces that are supported on the volume. Namespaces supported are DOS, Macintosh*, NFS*, FTAM, OS/2*, and NT.

- ◆ **Attributes:** Attributes of the volume. Possible attributes are block sub-allocation, file compression, data migration, auditing, and read-only. A volume can have a combination of attributes, such as read-only volume with block sub-allocation.
- ◆ **# Logical Segment:** The number of segments comprising this volume.
- ◆ **DS Name:** The volume's full Directory Services distinguished name or a zero-length string if not applicable.
- ◆ **Non-Purgable:** The amount of space (in kilobytes) taken by the deleted files whose purge dates have not yet expired. Non-purgable space can be reclaimed as free space when the deleted files become eligible to be purged.
- ◆ **Block Size:** The block size on the volume in bytes.
- ◆ **Dir Slots:** The total number of directory table entries available on the volume.
- ◆ **Used Dir Slots:** The number of directory table entries that are currently in use.
- ◆ **File System Name:** The type of file system on the volume is either remote or local. The File System Name value is listed only if the volume is remote. In this case, the file system name is the remote mount point; for example, SITE1:/usr/x.

Volume Trend View

You can select the Volume **Trend View** after expanding the following server objects: Services > File > Volumes > *volume_x*. This view provides the following graph:

- ◆ **Volume % Free Space:** The percentage of space still available on the volume.

Open Files View

The Volume Open Files View displays a table of all open files on the volume. If it is opened by more than one connection, multiple entries for the same file will appear in the table. You can select the Open Files View after expanding the following server objects: Services > File > Volumes > *volume_x*. This view provides the following information:

- ◆ **Filename:** The name of the open file, including the directory path.
- ◆ **Connection #:** The number of the connection that opened the file.
- ◆ **Login Name:** The name of the user (if any) who opened the file. If the file was opened by the system or by an NLM, the Login Name will be a zero-length string.
- ◆ **Directory Number:** A number that uniquely identifies an open file within a NetWare volume.
- ◆ **Volume ID:** A number that uniquely identifies a NetWare volume.

Volume Segment View

The Volume Segment View provides information about the segments on a volume. You can select this view after expanding the following server objects: Services > File > Volume > *volume_x*. As long as the Volume Segment View is displayed, the server is polled for data and the view is constantly updated with real-time information. This view provides the following information about each segment on the selected volume:

- ◆ **ID:** The number assigned to the volume segment for identification.
- ◆ **Logical Partition ID:** The number assigned to a logical partition for identification.
- ◆ **Physical Partition ID:** The number assigned to a physical partition for identification.
- ◆ **Size:** The size of the segment.

- ◆ **Fault Tolerance:** The type of fault tolerance used on the segment. Possible types are duplex and mirrored. If there is no fault tolerance, the value is None.
- ◆ **Disk Drive:** The name of the disk drive on which the segment resides.

Volume Usage View

The Volume Usage View provides information about the amount of volume space in use per user. As long as the Volume Usage View is displayed, the server is polled for data and the view is constantly updated with real-time information. You can select this view after expanding the following server objects: Services > File > Volumes > *volume_x*. This view provides the following information per volume user:

- ◆ **Used KB:** Number of kilobytes currently in use.
- ◆ **Limit KB:** Number of kilobytes to which a user is limited.
- ◆ **Username:** The user's login name.

Queues

You can display the following views of information about the NLM software on a managed server:

- ◆ [“Queues Summary View” on page 162](#)
- ◆ [“Queue Summary View” on page 162](#)
- ◆ [“Queue Trend View” on page 163](#)

Queues Summary View

The Queues **Summary View** provides the following information about the print queues on the managed server:

- ◆ **Queue Name:** The name of the queue.
- ◆ **Type:** The type of queue (for example, archive queue, job queue, or print queue).
- ◆ **# Jobs:** The number of print jobs in the queue currently.
- ◆ **# Print Servers:** The number of print servers serviced by the queue.
- ◆ **Volume:** The volume where the queue resides.
- ◆ **Add Job State:** Indicates whether or not the queue can add jobs.
- ◆ **Attach State:** Indicates whether or not the queue can attach.

Queue Summary View

The Queue **Summary View** provides the following information about the print jobs in the selected queue:

- ◆ **Job #:** A unique number assigned to the print job.
- ◆ **Position:** The print job's order in the print queue.
- ◆ **Bytes:** The number of bytes to be printed.
- ◆ **Description:** A description of the print job.
- ◆ **User:** The username of the user who submitted the job.
- ◆ **Entry Time:** The time the job was added to the queue.

- ◆ **Control Flags:** A value representing the control flags for the job. For example, some possible control flags are service auto start, execute, user hold, or operator hold.
- ◆ **Target Time:** The date and time the job is to be printed.
- ◆ **Target Server:** The target server for the job.
- ◆ **Actual Server:** The name of the server currently processing the job.

Queue Trend View

The Queues **Trend View** provides the following graph for each queue on the managed server:

- ◆ **Wait Time of Next Ready Job (sec):** The average length of time the next job waits in the queue.

Printers

You can get the detailed information about the printers installed in a managed server, including printer name, port, driver and description, status, error conditions, etc. You can display the following views of information about the processors on your managed servers:

- ◆ [“Printer Console View” on page 163](#)
- ◆ [“Printer Summary View” on page 163](#)

Printer Console View

You can access the Console View for the Printers object container after expanding the following server objects: Devices > Printers. This view displays the following information for each printer object in the container:

- ◆ Printer Name: Name of the printer

Printer Summary View

You can display the Summary View for an individual printer by expanding the following server objects: Devices > Printer > printer_x. This view displays the following information:

- ◆ Printer Name: The name of the printer
- ◆ Printer Status: The current status of this printer device. The status can be idle, printing, warm-up, or unknown state.
- ◆ Error Condition: The error conditions include lowPaper, noPaper, lowToner, noToner, doorOpen, jammed, offline, or serviceRequested.

Other Devices

From this view, you can get other devices like the keyboard and the mouse installed on a managed server.

- ◆ [“Other Devices on Console View” on page 163](#)

Other Devices on Console View

This displays other devices like the keyboard and the mouse.

The information about the keyboard includes:

- ◆ Keyboard Name

- ◆ Keyboard Type
- ◆ Driver Name
- ◆ Class
- ◆ Bus Type

The information about the mouse includes:

- ◆ Mouse Name
- ◆ Mouse Type
- ◆ Driver Name
- ◆ Class
- ◆ Bus type

Ports

From this view, you can install serial ports and parallel ports on a managed server.

- ◆ [“Ports Console View” on page 164](#)

Ports Console View

This displays information about the serial ports such as COM ports and parallel ports such as LPT ports. The information about the ports includes:

- ◆ Port Name
- ◆ Controller
- ◆ Bus Type

6

Using the MIB Tools

ZENworks® for Servers (ZfS) provides the tools to manage Simple Network Management Protocol (SNMP)-manageable devices on your network. This section describes the Management Information Base (MIB) tools, the SNMP MIB Compiler and the SNMP MIB Browser. It also explains how to set up and use the tools. See the following sections for more information:

- ◆ “Understanding MIB Tools” on page 165
- ◆ “Configuring MIBs and Setting Up MIB Tools” on page 174
- ◆ “Using the MIB Browser” on page 176
- ◆ “Maintaining MIBs” on page 184

Understanding MIB Tools

The following sections provide information about the tasks required for managing SNMP devices using the MIB Compiler and the MIB Browser.

- ◆ “About MIBs” on page 165
- ◆ “Understanding the SNMP MIB Compiler” on page 165
- ◆ “Understanding the SNMP MIB Browser” on page 166
- ◆ “Managing Devices with MIB Tools” on page 168
- ◆ “Trap Definitions” on page 169

About MIBs

To manage a device, you must obtain a copy of the MIB or MIBs that the device supports. A MIB is an ASCII text file, written in a precise format that describes the management information available on a particular class of devices. If, for example, you have an XYZ router from company X and you want to use ZfS for managing the router, company X must provide you with the XYZ router MIB. ZENworks for Servers provides many standard and vendor-proprietary MIBs, which are found in the MIB Pool folder in the MIB Server Pool folder. By default, ZfS compiles the most generally applicable of these MIBs.

If you want to compile any new MIBs, you must store them in the MIB Pool folder in the MIB Server Pool folder. The console user can select or remove MIB files from the MIB Pool folder in the MIB Server Pool folder. The MIB Compiler compiles the files listed in the MIB Pool folder in the MIB Server Pool folder.

Understanding the SNMP MIB Compiler

The MIB Compiler does the following:

- ◆ Parses a set of predefined SNMP MIB files written in ASN.1 and SNMP V1, V2 syntax and verifies their syntax.
- ◆ Stores the compiled files in the ZfS database, which lets all users access these compiled files from a central location.

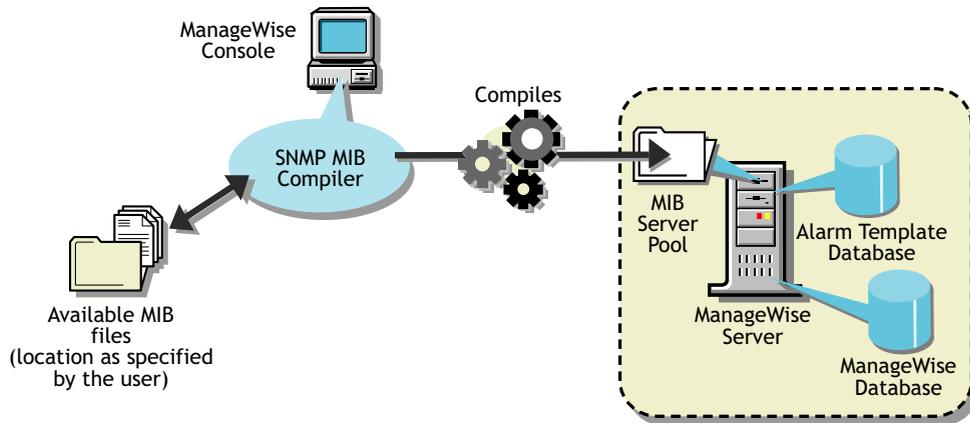
From the console, you can easily compile and maintain the MIB files located in the MIB Server Pool. You can add or remove MIB files from the MIB Pool.

- ◆ Updates trap definitions in the alarm template database.

The MIB Compiler lets you introduce new SNMP alarm templates into ZfS so they can be recognized and interpreted as alarms when they arrive at the console.

The Alarm Management System (AMS) interprets the annotations to trap definitions in a MIB to set the severity level and device status assigned to an alarm. The MIB files included with ZfS are already properly annotated.

The following figure demonstrates how the MIB Compiler incorporates information from the MIB files into the ZfS database:



During installation of ZfS, the MIB files that are precompiled using the MIB Compiler are also installed. The MIB for any SNMP node you want to manage must be compiled with ZfS. You can also integrate third-party MIBs. If you obtain a MIB file from a third-party vendor or any MIB file that was not installed with ZfS, you must compile the file using the MIB Compiler.

Using Role-Based Services with the MIB Compiler

ZfS role-based services let you assign various roles to users on your network. If your role is assigned the Enable MIB Compiler task, you can use the MIB Compiler.

See [“Role-Based Administration” on page 32](#) for more information about the role-based administration provided by ZfS.

Understanding the SNMP MIB Browser

The MIB Browser lets you manage SNMP-instrumented devices on the network.

To use this tool, you must have knowledge of SNMP and a good understanding of the structure of MIBs. Using the MIB Browser, you can manage nodes on the network by setting values of the MIB objects at the target nodes.

If you are familiar with the structure of an SNMP MIB, you can use the MIB Browser to retrieve data from SNMP-manageable node.

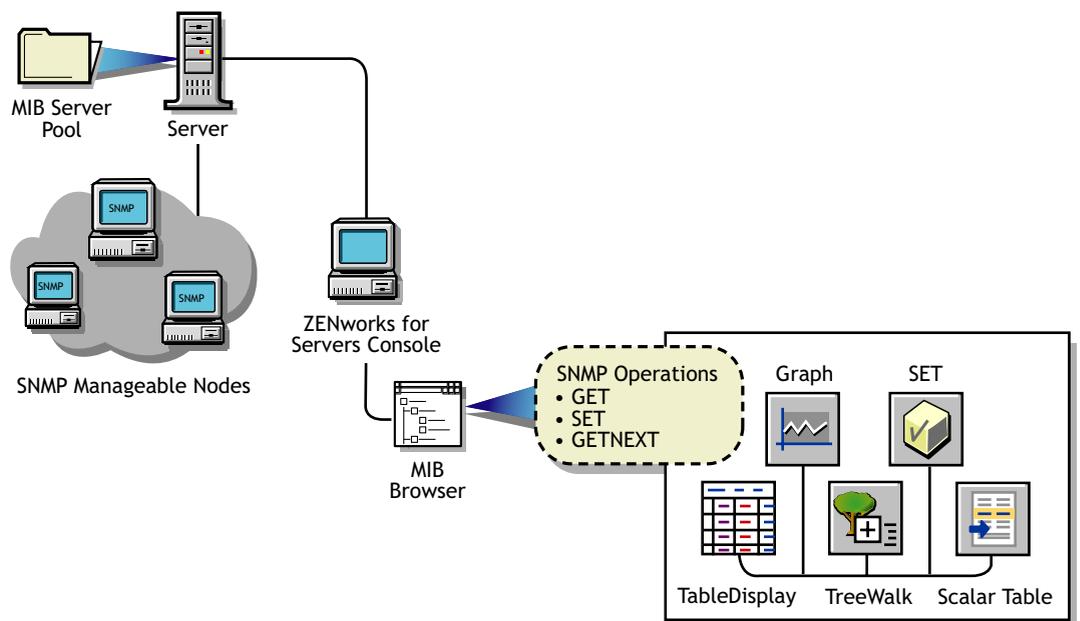
The MIB Browser lets you communicate with devices through an SNMP agent on the network over the User Datagram Protocol (UDP) or the Internet Protocol (IP). The results of SNMP commands are displayed in the MIB Browser window.

An SNMP agent is a program that provides access to management data about a particular network device and responds to SNMP Manager requests for the data. The NetWare® Management Agent software is an example of an SNMP agent that resides on a NetWare server. An SNMP agent resides in each manageable device on the network.

Although many ZfS windows display data retrieved from SNMP-manageable nodes, some administrators prefer the capability the MIB Browser provides for specifying the type of data they want to retrieve. Additionally, by using the MIB Browser, you can obtain some SNMP data that is not displayed in ZfS windows.

The MIB Browser takes the compiled MIB and displays the objects in a tree format. The MIB Browser also lets you walk the tree and look for the definitions of the selected MIB objects. You can set the community string to be used in the conversation between ConsoleOne® and the SNMP-manageable node to manage the device.

The following figure demonstrates the functionality of the MIB Browser:



The MIB Browser does the following:

- ◆ Represents the MIB information as a tree.

You can browse the objects in the MIB tree, which displays the composite OID (object identifier) for all compiled MIBs. The OID is the sequence of integers labeling each object on the path from the root of the tree to every object on the branches. The OID also describes the location of the object in the tree. For example, the novell(23) object in the tree is described as 1.3.6.1.4.1.23. For more information on the MIB tree, see [“Browsing the MIB Tree” on page 176](#).

- ◆ Retrieves specific information about the node using the SNMP GET and GETNEXT commands.

The MIB information is displayed as:

- ◆ A table display for tabular objects

You can add new rows to the table and issue SNMP SET commands to update the columnar values of the table. For more information, see [“Modifying Instances of an SNMP Table” on page 179](#).

- ◆ A graph display

If you choose to plot the SNMP requests, the Graph window displays the polled data of one or more MIB objects. For more information, see [“Graphing SNMP Request Results” on page 182](#).

- ◆ A scalar table display

You can form a scalar table by combining scalar objects. You can modify the scalar entries of the table. For more information, see [“Forming Tables of Scalar Objects” on page 181](#).

- ◆ A TreeWalk display

You can browse the OID values of scalar and tabular objects. For more information, see [“Viewing the Values of an Object and Its Child Nodes” on page 177](#).

- ◆ Changes the information at the target node using the SNMP SET command.

You can retrieve or change the value of MIB objects if the community strings match at the target node. The node should also allow remote setting of its variables.

- ◆ Creates a profile by saving the properties of the table, scalar table, or graph.

You open the profile to view a table, scalar table, or graph of different SNMP-manageable nodes on the segment with the properties specified in the profile. For more information, see [“Using a Profile for Tables and Graphs” on page 184](#).

For more information on the MIB Browser, see [“Using the MIB Browser” on page 176](#).

Using Role-Based Services with the MIB Browser

ZfS role-based services let you assign various roles to users on your network. If your role is assigned the Enable MIB Browser task, you can use the MIB Browser.

See [“Role-Based Administration” on page 32](#) for more information about the role-based administration provided by ZfS.

Managing Devices with MIB Tools

ZfS lets you manage any SNMP-manageable devices on the network. In particular, you can do the following:

- ◆ Set alarm templates for receiving alarms, often referred to as SNMP traps, for these devices
- ◆ Use the MIB Browser to display and set values on these devices

Before using the MIB Browser to manage the devices, you need to perform the following tasks:

1. Acquire the necessary MIBs.
2. Add trap annotations, if required.
3. Add or remove MIBs using the [MIB Compiler](#).

4. Run the MIB Compiler to compile the MIBs in ZfS.

ASN.1 and SNMP V2 Support

The MIB Compiler supports all MIB files written in ASN.1 and SNMP V1, V2 syntax. The MIB Compiler allows relaxation of ASN.1 syntax.

Trap Definitions

Some SNMP MIBs define the traps that a device can send to ConsoleOne when an unusual event occurs on the network. When you compile a MIB containing traps, information about those traps is added to the ZfS alarm database. When ZfS receives a trap, the information in the alarm database is retrieved and used by ZfS to generate the alarm summary string and to determine the alarm type, alarm severity, state of the affected device, and other details.

You can improve the presentation of the alarm information in ZfS by adding annotations to the trap definitions in the MIB files. These annotations are added as comments to the trap definitions so that the MIB compiles with third-party MIB compilers.

All Novell® MIBs are annotated. If you choose not to annotate the traps in other MIBs, ZfS displays the alarms; however, they are less readable. SNMP MIBs use the TRAP-TYPE macro to define traps.

This section covers the following topics:

- ◆ [“Keywords for Trap Definitions” on page 169](#)
- ◆ [“Template Database” on page 170](#)
- ◆ [“Keywords for Trap Annotations” on page 170](#)
- ◆ [“Example Trap Definitions” on page 171](#)
- ◆ [“Displaying Annotated Traps in ZENworks for Servers” on page 172](#)
- ◆ [“Formatting the SUMMARY String” on page 173](#)

Keywords for Trap Definitions

The following table explains a trap definition.

Keyword	Example	Explanation
TRAP-TYPE	duplpxNetAddr	Specifies the name of the trap. For example, duplpxNetAddr represents a duplicated IPX network address.
ENTERPRISE	netware-GA-alert-mib	Contains the OBJECT identifier of a node in the vendor’s tree, which, together with the trap number (the 8 following the ::= in DESCRIPTION) uniquely identifies the trap.
VARIABLES	(osName, osLoc, tiTrapTime, tiEventValue, tiEventSeverity, tiServer)	Defines an ordered sequence of MIB objects that are passed as parameters of the trap to provide additional information about the event. For example, osName is a text string specifying the name of the server sending the trap; osLOC is a text string specifying the location of the server; tiTrapTime is an integer specifying the time the event occurred.

Keyword	Example	Explanation
DESCRIPTION	"Two servers use the same IPX Internet address."	Provides a textual description of the semantics of the trap.
Trap_number	:=8	Defines the trap.

Template Database

The MIB Compiler populates the alarm template database with the trap definitions in the MIB files. Any traps from the agents are stored in the database.

Keywords for Trap Annotations

The following table lists and explains the keywords you can use to annotate traps:

Keyword	Explanation
--#TYPE	Short name for the alarm. The name can contain a maximum of 40 characters. If this annotation is not present, the SNMP trap name is used. Every trap should have a unique type.
--#SUMMARY	Description of the alarm with placeholders and formatting information for the actual parameters passed with the alarm. See "Formatting the SUMMARY String" on page 173 for more information. Without this annotation, the alarm summary string lists each SNMP parameter name followed by its value.
--#ARGUMENTS	List of parameters to substitute in the SUMMARY string. Parameters are substituted in the order in which they appear in the list. Each element of the list is the index (zero-based) of the parameter in the VARIABLES clause.
--#SEVERITY	Default severity assigned to the trap. This can be one of the following: <ul style="list-style-type: none"> ♦ INFORMATIONAL ♦ MINOR ♦ MAJOR ♦ SEVERE ♦ UNKNOWN Alarms with a default severity set to SEVERE are displayed in the ticker tape. Without this annotation, the severity is displayed as UNKNOWN.
--#TIMEINDEX	Index of the variable in the VARIABLES clause. This index contains the time when the alarm was generated. The time is expected to be an integer representing the number of seconds since 1970 (UNIX* time). If such a variable does not exist in the VARIABLES clause, use an index greater than the total number of variables in the VARIABLE clause.
--#HELP	This index contains name of the help file.
--#HELPTAG	The index contains the reference to the Help ID of the help file that is specified in the HELP index.

Keyword	Explanation
--#STATE	<p>Default state of the object when the alarm was generated. This can be one of the following:</p> <ul style="list-style-type: none"> ♦ OPERATIONAL ♦ NONOPERATIONAL ♦ DEGRADED ♦ UNKNOWN <p>Without this annotation, the state is UNKNOWN.</p>

Note the following rules about adding trap annotations:

- ♦ Each annotation must be embedded in a comment. Everything from the double hyphen to the end of the line is treated as a comment.
- ♦ Each annotation must be on a separate line.
- ♦ Annotations must appear in the order in which they are discussed in [“Trap Definitions” on page 169](#).
- ♦ All annotations must be inserted after the DESCRIPTION clause and before the ::= clause.
- ♦ STATE and SEVERITY values are written to the alarm database the first time the MIB is compiled. If you want to modify the STATE and SEVERITY values for the alarm templates, modify these values in the corresponding MIB files and recompile using the MIB compiler.

Example Trap Definitions

The following sections explain a trap description in an SNMP trap before and after annotation:

- ♦ [“Example Trap Definition Before Annotation” on page 171](#)
- ♦ [“Example Trap Definition After Annotation” on page 172](#)

Example Trap Definition Before Annotation

```
dupIPXNetAddr TRAP-TYPE
ENTERPRISE netware-GA-alert-mib
VARIABLES {osName, osLoc, tiTrapTime, tiEventValue, tiEventSeverity, tiServer}
DESCRIPTION "Two servers use the same IPX internetwork address."
::=8
```

Example Trap Definition After Annotation

```
dupIPXNetAddr TRAP-TYPE
ENTERPRISE netware-GA-alert-mib
VARIABLES {osName, osLoc, tiTrapTime, tiEventValue, tiEventSeverity, tiServer}
DESCRIPTION "Two servers use the same IPX internetwork address."
::=8
```

– Trap annotations are as follows:

```

--#TYPE "Duplicate IPX address"
--#SUMMARY "%s at %s and %s are using the same IPX address"
--#ARGUMENTS {0,1,5}
--#SEVERITY CRITICAL
--#TIMEINDEX 2
--#HELP "MYHELP.HLP"
--#HELPTAG 60004
--#STATE DEGRADED
::=8

```

Displaying Annotated Traps in ZENworks for Servers

Assume that the dupIpxNetAddr trap shown in [“Keywords for Trap Definitions” on page 169](#) was received by ZfS with the following variables:

- ◆ osName = SJM-JACK
- ◆ osLoc = JACK’s CORNER
- ◆ tiTrapTime = ~700000000
- ◆ tiServer = SJM-TIM

To display a trap, use the Active Alarm, Alarm History, or Alarm Detail window. The following example shows the result:

Receive Time:03/04/99 09:15:45

Alarm Type: Duplicate IPX address

Summary: SJM-JACK at JACK’s Corner and SJM-TIM are using the same IPX address

Severity: Critical

State: Degraded

When you select the alarm on the Alarm Report table and click the NetWare Expert button, ZfS displays help information for this alarm.

Formatting the SUMMARY String

The SUMMARY keyword in the trap annotation lets you provide the actual wording of the alarm summary. This wording is used by ZENworks for Servers when the alarm occurs.

Placeholders within the string are replaced by actual parameters of the trap before the string is displayed by ZENworks for Servers. Each placeholder format string begins with a percentage sign (%) and tells ZENworks for Servers how to format the parameter that will be substituted for the placeholder in the final string. See [“Trap Definitions” on page 169](#) for a list of all available format strings for each parameter type and the printed form for each value.

The placeholder format strings are substituted, in order, by the parameters specified in the ARGUMENTS keyword. The ARGUMENTS keyword lists the (zero-based) index of each trap parameter as specified in the VARIABLES clause. The indexes are listed in the order in which you want them to be substituted in the SUMMARY string.

ZENworks for Servers can display a maximum of 140 characters in the SUMMARY string. Use the characters to display the most relevant information about the alarm. If you have a long SUMMARY string and want to keep the line length of the MIB file reasonable, you can insert multiple, consecutive SUMMARY annotations and the strings will be concatenated. For example, the following annotations below yield the same string:

–#SUMMARY “%s at %s and %s are using the same”

–#SUMMARY “IPX address”

–#SUMMARY “%s at %s”

–#SUMMARY “and %s are”

–#SUMMARY “using the same IPX address”

The following table lists the format strings and parameter types.

Parameter Type	Format String	Printed Form
BOOLEAN	%s	True or False.
	%d	1 or 0.
INTEGER	%x	HEX.
	%d	DECIMAL.
	%t	Prints the integer or a date and time (Greenwich Mean Time). The integer represents seconds since 1970.
OCTET STRING	%s	Prints the text string with all control characters taken out.
	%m	Prints the first 6 bytes of data as a hyphen-separated MAC address. For example, 00-00-07-00-07.
	%x	Prints the octet string in hexadecimal. For example, 0000070007.
NULL	%d	Prints the number 0.
	%s	Prints the string NULL.
OBJECT IDENTIFIER	%s	Prints dot-separated decimal values. For example, 1.3.6.5.4.
IP Address	%s	Prints dot-separated IP address. For example, 13.56.56.56.
	%x	Prints a long hexadecimal value.
BIT STRING	%s	Prints each byte as decimal.

Configuring MIBs and Setting Up MIB Tools

This section describes the procedural tasks for configuring MIBs and setting up the community strings for SNMP operations on an individual node. After you complete these tasks, you can perform SNMP operations using MIB Tools.

This section covers the following topics:

- ◆ [“Annotating Third-Party MIBs for Integration with ZfS” on page 174](#)

- ◆ [“Compiling MIBs for SNMP-Manageable Nodes” on page 175](#)

Annotating Third-Party MIBs for Integration with ZfS

When you compile a MIB containing SNMP traps (alarms), information about those traps is added to the ZfS alarm database. This information can then be displayed in ConsoleOne.

All Novell MIBs are annotated so that the alarm information displayed in ConsoleOne is easily readable. This alarm information includes a summary describing the alarm, the alarm severity, and the state of the affected node. Third-party MIB files do not necessarily contain this same information. Therefore, the information about the traps in third-party MIBs is not as meaningful when displayed in ConsoleOne.

You can add annotations to third-party MIB files for the trap definitions so that the alarm information displayed in ZfS for those traps is more readable than if you compile the MIB as is. Any annotations you add to a third-party MIB are added as comments to the trap definitions. This ensures that the MIB still compiles with third-party MIB compilers.

If you do not annotate the traps in third-party MIBs, ZfS will display the alarms. The MIB Compiler displays warnings in the status display about the missing annotations.

To add annotations to a third-party MIB:

- 1** Open the MIB in a text editor.
- 2** Add any of the annotations shown in [“Keywords for Trap Annotations” on page 170](#), by following these rules:
 - ◆ Enter annotations only between the DESCRIPTION and the "::-=" clause.
 - ◆ Each annotation must be on a separate line.
 - ◆ Annotations must be in the order shown in [“Keywords for Trap Annotations” on page 170](#).
 - ◆ Embed each annotation as a comment. Precede each annotation with two hyphens and a pound sign (#).
For example: `--#Type "type_description"`
For a full example, see [“Example Trap Definitions” on page 171](#).
- 3** When you finish annotating trap definitions, save your changes and exit the text file.
Compile the MIB, as described in [“Compiling MIBs for SNMP-Manageable Nodes” on page 175](#).

Use ConsoleOne Alarm Disposition table to view the values for the alarm severity level and alarm state from the default values in the SNMP MIBs. If you change the value for an alarm’s severity or state after you compile the MIB, you must recompile the MIB for those changes to overwrite any changes made through the Alarm Disposition table.

Compiling MIBs for SNMP-Manageable Nodes

The MIB Compiler lets you manage the MIB Server Pool and also compile the .MIB files contained in the MIB Server Pool. The information in the compiled files is placed in the database on the ZfS server. The MIB Browser and the SNMP protocol decoder use this database.

The MIB Compiler also adds or updates any trap definitions to the alarm template database for use by the ZfS Alarm Management System (AMS).

The MIB Server Pool contains the list of MIB files. You can add or remove the MIB files from the MIB Server Pool.

To compile the MIBs:

- 1 From ConsoleOne, click the ZfS server node.
- 2 Right-click the node > click Properties > click the MIB Pool tab.

The current MIB Pool lists the compiled MIB files present in the database.

- 3 Choose your options.
 - ◆ To add MIBs, click Add to locate the .MIB files and add them to the MIB Pool list.
The added MIBs are displayed in the adjacent list box.
When you add MIBs, you choose to integrate or exclude the trap information while compiling MIBs. If you do not integrate traps with the MIBs, only the MIB information is stored in the database on successful compilation of the MIBs. Click Advanced > select the Trap Integration check box to integrate the trap information with the MIBs.
 - ◆ To remove files from the MIB Pool list, select the MIB from the list > click Remove.
 - ◆ To compile the MIBs with less strict adherence to ASN.1 syntax, click Advanced > select the ASN.1 Syntax Relaxation option.

- 4 Click Compile.

The MIB Compiler compiles all files in the MIB Pool list with the .MIB extension and updates the database. The compilation process is begun by launching a Results dialog box. This dialog box displays the status information of the MIBs including the MIBs that were successfully compiled, MIBs that were not compiled and the corresponding error message, and the status of updating the database with the MIB compile information, and the status of updating the Alarm database.

IMPORTANT: You cannot closed the Results dialog box during compilation. The Close button in the Results dialog box is disabled during compilation. You can close this dialog box only after the compilation is successful or failed.

- 5 Click Close.

IMPORTANT: If the SNMP MIB is not set up correctly, or an imported Request for Comments (RFC) is not available during compilation of the MIB, or any other .MIB file is not available, an error message is generated in the MIB Compiler window. Add the required RFC or the dependent MIB and compile.

Using the MIB Browser

This section acquaints you with using the MIB Browser to manage SNMP-manageable nodes.

This section includes the following topics:

- ◆ “Browsing the MIB Tree” on page 176
- ◆ “Viewing the Values of an Object and Its Child Nodes” on page 177
- ◆ “Configuring a Node by Setting Object Values” on page 178
- ◆ “Modifying SNMP Preferences” on page 179
- ◆ “Modifying Instances of an SNMP Table” on page 179
- ◆ “Forming Tables of Scalar Objects” on page 181
- ◆ “Graphing SNMP Request Results” on page 182

- ◆ “Using a Profile for Tables and Graphs” on page 184

Browsing the MIB Tree

The MIB Browser lets you select the objects you want to display, and it sends SNMP queries to the node to obtain the data objects that you requested. It also allows SNMP operations such as GET, GETNEXT, and SET requests on a particular object in the MIB of an SNMP-managed node.

The MIB Browser periodically polls the node and continually updates the display. You can view and modify scalar and tabular data objects.

MIB Tree Browser

Within the MIB browser, the MIB Tree Browser is a graphical display of management data that consists of numerous objects.

The MIB Browser displays a composite OID tree for all compiled MIBs. Analogous to a file system, the MIB Browser shows leaf objects, which are the SNMP data objects.

The MIB Browser spans the selected node with its subtree and leaf objects and displays the name of the objects in the MIB Tree Browser. You browse from the highest level of the tree and view the leaf object values.

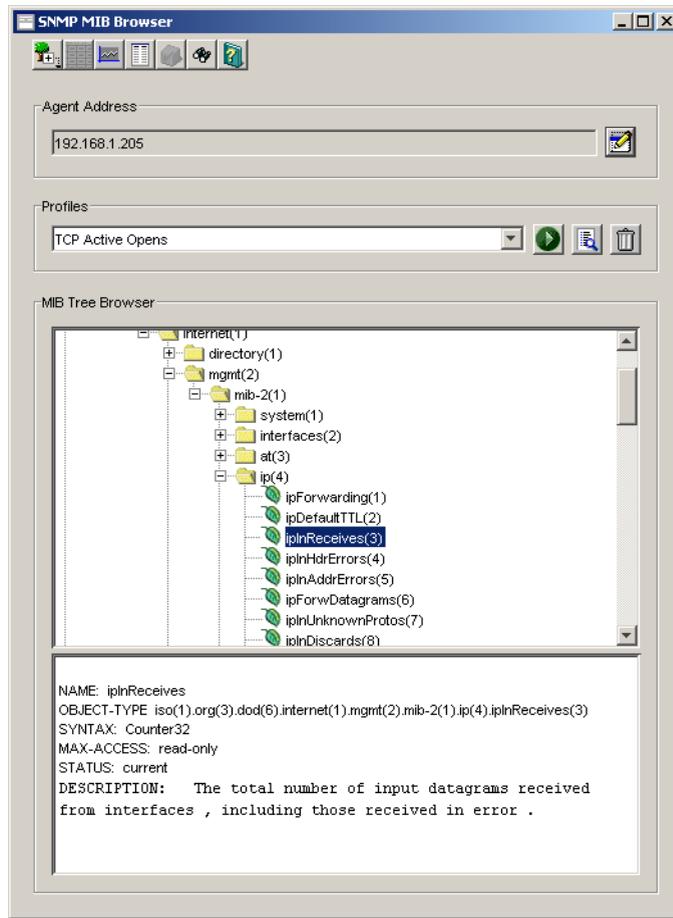
The top pane of the MIB Tree Browser displays the tree with the selected object. Each object is displayed as a file folder icon, followed by its SNMP name with the SubId appended in parentheses. If the object is a non-leaf node, the MIB Tree Browser also displays its children.

The bottom pane describes the selected object. The description is derived from the compiled MIB file. The format of the description is as follows: textual description of the object, full numeric OID and object name, ASN.1 type, size, textual convention, access, Index clause taken from the Entry object, status, and description.

For example, for an internal node SYSTEM with child nodes, the child nodes describe the properties of the SYSTEM. The OID of SYSTEM is iso(1).org(3).dod(6).internet(1).mgmt(2).mib-2(1).system(1). Another equivalent representation of this OID is 1.3.6.1.2.1.1. Note that the parent node does not have information, and the child nodes contain the properties.

The child nodes of SYSTEM are sysDescr OID(1.3.6.1.2.1.1.1), sysObjectID OID(1.3.6.1.2.1.1.2), sysUpTime OID(1.3.6.1.2.1.1.3), sysContact OID(1.3.6.1.2.1.1.4), sysName OID(1.3.6.1.2.1.1.5), sysLocation OID(1.3.6.1.2.1.1.6), and sysServices OID(1.3.6.1.2.1.1.7).

The following figure shows the MIB Browser window.



To browse the MIB objects:

- 1** From ConsoleOne, click the target SNMP-manageable node.
- 2** Click File > Action > MIB Browser.
- 3** Click the object whose values you want to view from the MIB Tree Browser.
 - ◆ To select an object, click the name text or the icon in the MIB Browser tree.
 - ◆ To expand or collapse the next level in the tree display, double-click the object.

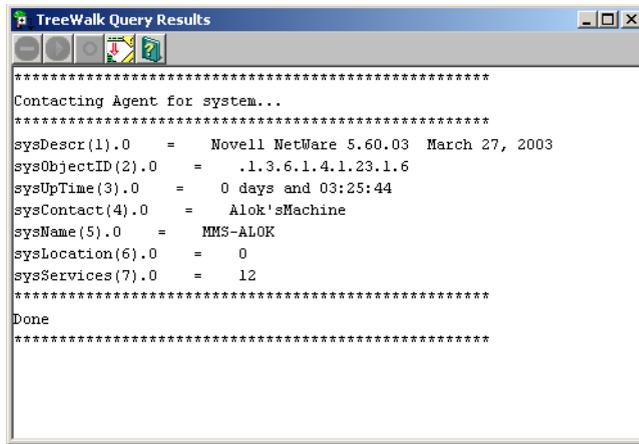
Viewing the Values of an Object and Its Child Nodes

The MIB Browser spans the selected node with its subtree and leaf objects and displays its values in the TreeWalk Query Results window. You can browse the OID values of scalar and tabular objects.

To view the values of the instances of a MIB object:

- 1** From ConsoleOne, click the target SNMP-manageable node.
- 2** Click File > Action > MIB Browser.
- 3** Click the object > Perform TreeWalk for the node button.

The following figure shows the TreeWalk Query Results window.



If you select a leaf object, you can view the values for each instance of this object. For non-leaf objects, this window will display all the values of the child node of this object. For example, if you want to view the values of the child nodes for the object *system*, click the parent object *system*.

The display process in the TreeWalk Query Results window continues recursively for all the non-leaf objects of the selected object. You can pause and resume this display in the window.

Customizing the Display of TreeWalk Query Results Window

The TreeWalk Query Results window displays the number of lines based on the settings specified in the TREEWALK.PROPERTIES file.

This file, located in `\CONSOLEONE\version\BIN\SAVED-VIEWS\GENERIC` directory, contains the following setting:

MaximumNumberOfLine=number_of_lines_for_display

where *number_of_lines_for_display* is the number of lines that will be displayed at a time. The default setting is 10,000 lines. You can modify this setting. The settings will apply only if you restart ConsoleOne and bring up the TreeWalk Query Results window. To clear the display in the TreeWalk Query Results window when the text buffer is full, click the Clear button. There may be some out of memory problems if you specify a large line setting in the TREEWALK.PROPERTIES file.

Configuring a Node by Setting Object Values

Using the MIB Browser, you can issue an SNMP SET command to change information at an SNMP-manageable node if you have the appropriate privileges. You select a scalar object from the MIB Browser and set its value.

You can modify the values for an integer, enumerated integer, object identifier, string, and IP address object types.

To issue an SNMP SET command for a scalar object:

- 1** From ConsoleOne, click the target SNMP-manageable node.
- 2** Click File > Action > MIB Browser.
- 3** Click a scalar object whose values you want to view > click Display Data As a Scalar Table.
- 4** Specify the object value for the scalar object.
- 5** Click OK.

To modify columnar values of an SNMP table, see [“Modifying Instances of an SNMP Table” on page 179](#).

Modifying SNMP Preferences

SNMP parameters are used to communicate with the target device. The MIB Browser lets you change the SNMP community strings or specify the transport address of a new target device.

Any SNMP operation requires these values to be set. After starting an SNMP operation, such as polling a table, changing the SNMP preferences does not affect the operation.

You can modify the following parameters:

Agent Address: You can specify the IP or IPX address and the Domain Name System (DNS) name of the SNMP-manageable node to which you want to send an SNMP request. This node should have an SNMP agent.

SET and GET Community Strings: The community string that ZfS uses must match the one expected by the SNMP agent in the managed node or the SNMP operations will fail. If the SNMP agent on the node expects a community string for SET and GET operations that is different from public (the default), you can specify the expected community string to override the default community or those community strings you set previously. You can use Unicode* or International characters for the community string.

To modify the SNMP preferences:

- 1** From ConsoleOne, click the target SNMP-manageable node.
- 2** Click File > Action > SNMP MIB Browser.
- 3** Click Modify SNMP Preferences.
- 4** Specify the parameters > click Close.

Modifying Instances of an SNMP Table

A table in an SNMP MIB is an SNMP construct derived from the structure of the MIB. Each row in the table corresponds to a row in the SNMP table.

The MIB Browser provides the Table Display window to display tabular objects you select. This window displays one or more rows from an SNMP table in a two-dimensional grid and follows the SNMP index order to display rows.

The table shows each column in the SNMP table as columns. Each column heading is derived from the SNMP table columns. The Table Display window displays the columns with their values as single or multiple rows for the MIB you selected.

SNMP allows operations on individual table entries only. The OID identifies the column and row.

From the MIB Browser, you can perform the following operations:

- ◆ Add rows to an SNMP table
For more information, see [“Adding Rows to an SNMP Table” on page 180](#).
- ◆ Modify a row of an editable table
For more information about adding or modifying rows, see [“Adding Rows to an SNMP Table” on page 180](#).

- ◆ Save the table as a profile

For more information about saving a table as a profile, see [“Using a Profile for Tables and Graphs” on page 184](#).

Viewing the SNMP Table

To view the SNMP table:

- 1 From ConsoleOne, click the target SNMP-manageable node.
- 2 Click File > Action > SNMP MIB Browser.
- 3 Click a tabular object whose values you want to view > Display Data As a Table.

From the Table Display window, you can add rows or modify the rows of the SNMP table and input values for each column. For more information about adding or modifying rows of an SNMP table, see [“Adding Rows to an SNMP Table” on page 180](#).

The following figure shows the MIB Browser Table Display window.

hostTopNReport	hostTopNIndex	hostTopNAddress	hostTopNRate
6,300	1	FF FF FF FF FF FF	497
6,300	2	00 90 21 5A BC 27	270
6,300	3	00 03 47 1F 3D 2F	219
6,300	4	01 80 C2 00 00 00	49
6,300	5	01 00 5E 00 01 16	29
6,300	6	01 00 5E 7F FF FD	26
6,300	7	01 00 81 00 01 00	10
6,300	8	01 00 81 00 01 01	10
6,300	9	01 00 1D 00 00 00	10
6,300	10	03 00 00 00 00 01	6
6,300	11	00 B0 D0 22 C3 72	4
6,300	12	00 00 1D 29 19 FF	1
6,300	13	00 08 C7 C9 88 C2	1
6,300	14	01 00 0C CC CC CC	1
6,300	15	00 C0 4F 37 C6 93	1
6,300	16	00 E0 2B 00 00 00	1
6,300	17	00 0F 20 33 46 59	1
6,300	18	00 01 03 21 A7 8D	0
6,300	19	00 0B CD 10 C5 86	0
6,300	20	08 00 20 9A 0E 5E	0

The MIB Browser periodically sends SNMP queries to the node to obtain the data objects you request. When you provide new values for writable objects, the MIB Browser writes these values to the node. The MIB Browser periodically polls the node and continually updates the display. You can change the polling interval by suspending the SNMP interaction or by canceling the SNMP interaction.

Adding Rows to an SNMP Table

When you add a new row to an SNMP Table, the MIB Browser generates the SNMP SET request.

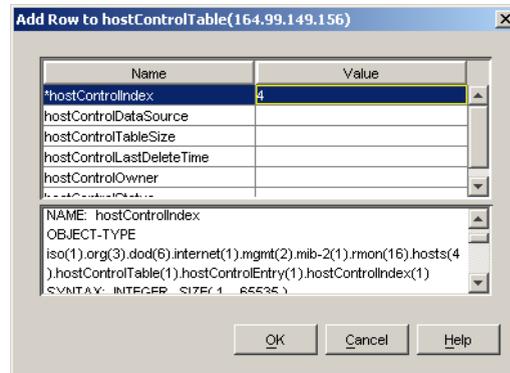
Before generating the SNMP Set request, the MIB Browser sends a GET command to the node that you selected in the MIB Browser table and retrieves the value of the object. On adding rows with the specified values for the objects, the MIB Browser issues multiple SNMP SET commands to update the SNMP table.

To add a row to an SNMP table:

- 1 Click the table object from the MIB Browser window > Add a New Row to the Table.

For more information about selecting the table object, see “[Modifying Instances of an SNMP Table](#)” on page 179.

The following figure shows the Add Row to Table window.



- 2 Double-click the row.
- 3 Modify the value > click OK > click OK.

To add rows in an SNMP table, you must input the values for all the index rows, which are denoted by asterisks.

To modify a row of an editable table:

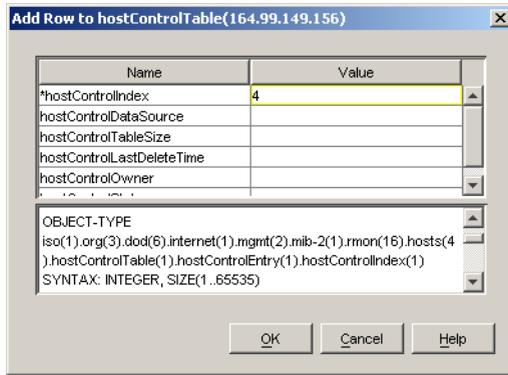
- 1 Open the Table window.
- 2 Click the row whose values you want to modify > click Issue SNMP Set request for a column button.
- 3 Double-click the row.
- 4 Modify the value of the object > click OK > click OK.

Forming Tables of Scalar Objects

You can make a scalar table by combining the scalar objects from the MIB Browser. A scalar table is a two-column table with the name and value of the scalar object entries. To create a scalar table, you select a group node with scalar child nodes or a group of scalar objects. For example, you add one or more scalar objects such as ipInDelivers and SysUpTime to make a new scalar table labeled ipInDeliversTable.

If you want to view the scalar tables that you create, save the scalar table as a profile. You can load the scalar table profiles when required.

The following figure shows the Scalar Table window.



To combine scalar objects as a scalar table and view the table:

- 1** Create a new scalar table.
- 2** Add to or modify the existing table by adding scalar entries or by removing entries from the table.
- 3** Save the scalar table as a profile.
- 4** Launch the profile.

To create a new scalar table:

- 1** From ConsoleOne, click the target SNMP-manageable node.
- 2** Click File > Action > SNMP MIB Browser.
- 3** Right-click a scalar group or a scalar object > click New > click Scalar Table.

To add or remove scalar entries to an existing table:

- 1** Open an existing scalar table.
- 2** Toggle to the MIB Browser window > click Add to > click *Scalar_table_name*.

Alternatively, click the scalar entry in the MIB Browser window, and from the Scalar Table window, click Add Node Selected from Browser Window.

To remove the scalar entry, click the scalar entry in the Scalar Table window, and click Remove the Node Selected in This Window.

Graphing SNMP Request Results

You can plot the SNMP request results in a graph that displays the polled data of the MIB objects. Only attributes of ASN.1 type Integer, Counter, Time Ticker, and Unsigned Integer are plotted as current absolute values.

You can plot more than one object in the same graph, add more objects, or remove the MIB objects from the existing graph. If you want to view the graphs that you create, save the graph as a profile. You can then load the graph profiles when required.

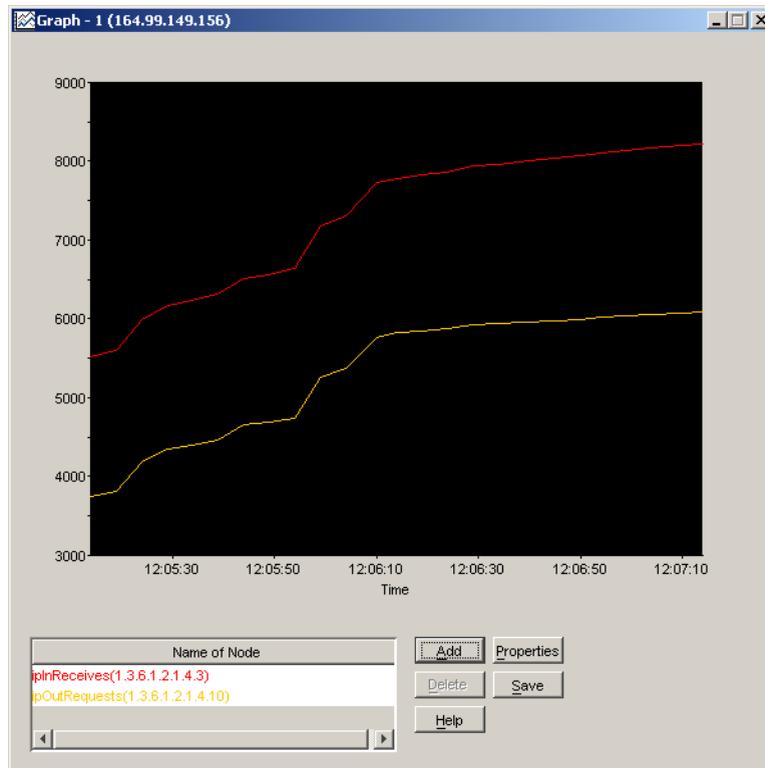
To graph SNMP request results of one or more nodes:

- 1** Click the target SNMP-manageable node from the console.
- 2** Click File > Action > SNMP MIB Browser.
- 3** Click the MIB object whose values you want to plot.

4 Right-click the object > click New > click Graph.

The MIB Browser plots the graph with the values of the selected object and its leaf object values dynamically in the Graph pane of the window.

The following figure shows the Graph window.



To graphically plot the values of more than one object:

- 1** Toggle to the MIB Browser window.
- 2** Click the MIB object you want to plot > click Add To > click the Graph.
You add these objects to any of the active graph windows you want.

Alternatively, you can click the MIB object from the MIB Browser window and then click the Add button in the MIB Browser Graph. Remove the objects from the list that you do not want by selecting the node from the list and clicking the Delete button.

From the Graph window, you can perform the following operations:

- ◆ Rescale the Y-axis of the graph
- ◆ Set the period to display
- ◆ Set the polling interval and refresh rate of the display

By default, the values plotted in the graph are absolute. If you want to view the rate of change of values per second with respect to sysUpTime, you must click the Rate option. For example, if you click ipInPackets and choose the Rate option, you can view the values per second.

Using a Profile for Tables and Graphs

A profile contains information about the properties of the graph, table, or scalar table. You use a profile to specify the information, such as the method of display (table or graph) and polling interval.

You create a profile by saving the properties of the table, scalar table, or graph as a profile. You open the profile to view a table, scalar table, or graph of different SNMP-manageable nodes on the segment with the same properties specified in the profile. You can modify or delete the profile.

To form a profile:

- 1** Save the properties of the display window.
- 2** Open the profile.
- 3** Modify the properties of the profile as required.

To save a profile:

- 1** Click the Save button from the Scalar table window, Graph window, or Table window.
- 2** Type the details of the profile.
Specify the name, description, and properties of the objects.
- 3** Click OK.

To open a profile:

- 1** From the MIB Browser window, click the profile you want from the drop-down list.
- 2** Click Launch This Profile.

To modify the selected profile:

- 1** Click View/Edit Profile Contents for the selected profile in the MIB Browser window.

To delete a selected profile:

- 1** Click Delete This Profile.

Maintaining MIBs

Depending on your need to add MIBs for managing nodes, you must compile the MIBs.

To delete a particular MIB from ZfS, remove the appropriate MIB text file from the MIB Server Pool and rerun the MIB Compiler. If the MIB you delete contains traps, you must remove the alarm definitions before you rerun the MIB Compiler.

When you add MIBs, you choose to integrate or exclude the trap information while compiling MIBs. If you disallow trap integration with the MIBs, only the MIB information is stored in the database on successful compilation of the MIBs.

For more information about how to add or remove MIBs, refer to [“Compiling MIBs for SNMP-Manageable Nodes” on page 175](#).

7

Monitoring Services

ZENworks® for Servers (ZfS) lets you test the connectivity and availability of a service on a network device. This test checks and measures the response by sending diagnostic packets, and also notifies the console whenever the status of the service changes.

This section provides an overview of the testing facility, lists the services that can be monitored on the nodes, and discusses the test options. See the following sections for more information:

- ◆ “Understanding Monitoring Services” on page 185
- ◆ “Monitoring Services on Target Nodes” on page 187

Understanding Monitoring Services

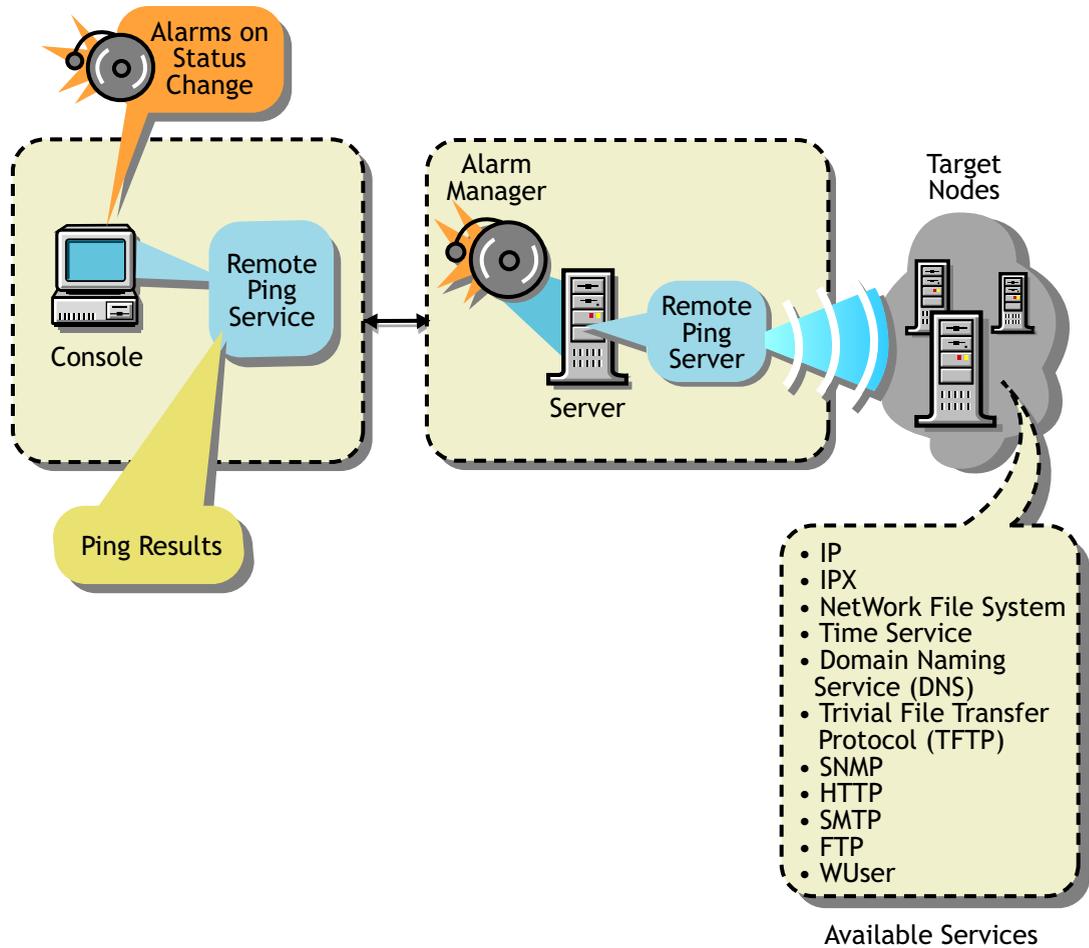
Using the Monitoring Services facility, you test connectivity of services on one or more critical network devices, such as servers or routers. For example, you can monitor services because you want to be alerted immediately if the connectivity between the console and critical nodes is disrupted.

This test facility enables testing of the following services:

- ◆ Domain Name System (DNS)
- ◆ Dynamic Host Configuration Protocol (DHCP)
- ◆ Echo
- ◆ File Transfer Protocol (FTP)
- ◆ Hypertext Transfer Protocol (HTTP)
- ◆ Hypertext Transfer Protocol Secure (HTTPS)
- ◆ Internet Packet Exchange™ (IPX™)
- ◆ Internet Protocol (IP)
- ◆ Network File System (NFS)
- ◆ Network News Transfer Protocol (NNTP)
- ◆ Simple Mail Transfer Protocol (SMTP)
- ◆ Simple Network Management Protocol (SNMP)
- ◆ Time Service
- ◆ Trivial File Transfer Protocol (TFTP)
- ◆ WUser

The test facility uses the ZfS server as the remote ping server. When you select the service on the node for testing, the console interacts with the remote ping server on the ZfS server and displays the results of the test on the console.

The following figure shows a graphical representation of Monitoring Services.



To monitor nodes, you choose the nodes and enable the monitoring session for the duration you require.

From the console, you monitor the services in the following ways:

- ◆ Test connectivity of the services on a node one time only when you suspect a problem with the connectivity.
- ◆ Continuously monitor connectivity of the services on a critical node until you close the test facility.
- ◆ Continuously poll the services of the nodes on the segment (for example, connectivity testing of the services on the target nodes runs uninterrupted until you disable monitoring). If you do not disable monitoring, this test facility continues even after you close the console.

For testing connectivity of services on the target nodes you select, you set the following options:

- ◆ Specify the services on the selected target nodes.
If you need to test any TCP-based services, add the service to the existing list of services.
- ◆ Define the test interval between two successive tests.
- ◆ Define the timeout value.

The timeout value determines the time duration that the remote ping server waits to receive the response from the target node.

You can view the status of the connectivity and measure diagnostics, such as round trip delays or number of packets sent and received from the console.

Role-Based Services for Using the Monitoring Services

Role-based services (RBS) defines the task for Monitoring Services as Enable Remote Ping. If this task is assigned to your role, you can use the Monitoring Services facility.

For general information about role-based traffic analysis tasks, creating RBS role objects or specifying tasks that RBS roles can perform, see [“Role-Based Administration” on page 32](#).

Monitoring Services on Target Nodes

This section guides you through the tasks involved in using the Monitoring Services facility.

From the console, you can monitor critical nodes on the network and manage potential connectivity problems before they affect the network. You define the services to test on the selected nodes, then view the test results and other data for each listed target. To perform the testing, complete the following general steps:

1. Define the targets to be monitored.

See [“Defining the Targets for Monitoring Services” on page 187](#) for information about specifying the services on the target nodes.

2. On a per-node basis or on multiple nodes, change the test interval or timeout value.

These tests use default values for the test interval between two successive tests on the target and to determine the time duration that the remote ping server waits to receive the response from the target node. You can change these values for the test.

See [“Changing the Test Options for a Node” on page 190](#) for information about editing the test options.

3. View the test results.

The nodes are monitored continuously, at the defined test interval for the node. Depending on the Monitoring Services test that you choose, the corresponding test results are displayed.

See [“Displaying Test Results Data” on page 189](#) for information about test results data.

Defining the Targets for Monitoring Services

Monitoring Services requires that you specify the targets for the tests. You can choose from the following test options:

- ♦ [“Test the Services on the Target Node One Time Only” on page 188](#)
- ♦ [“Continuously Monitor the Services on the Target Nodes” on page 188](#)
- ♦ [“Continuously Poll the Services of the Target Nodes on a Segment Until the Test Is Disabled” on page 188](#)

NOTE: You can monitor approximately 50 critical services simultaneously on the servers. Monitoring more than 50 services may overload the server memory and result in performance degradation.

Test the Services on the Target Node One Time Only

If you suspect a problem with a node in the network, you can ping the node once for monitoring services. When you select the target node for testing the services and specify the IP or IPX address, this address will determine the service that will be tested at the node. For example, if you type the IPX address, the default IPX service is tested on the target node.

The results of the test will display the status of the target node and details of the round trip delay in the Ping window.

To test the services on a node once:

- 1 From ConsoleOne, right-click the selected node > click Ping.
- 2 Type the ping target details.
- 3 Click OK.

Continuously Monitor the Services on the Target Nodes

To specify the services for continuous monitoring, add the targets and choose the services on the node and other options. The target node will be added to the list of targets in the Connectivity Test Results window and the test results data will be displayed. Monitoring of services continues until you close this window.

To define the targets for testing services on the node:

- 1 From ConsoleOne, click Action > Connectivity Test.
- 2 Click Add.
- 3 Specify the details for the target nodes in the Add Ping Target dialog box.
Refer to [“Adding Services for Monitoring” on page 190](#) for more information about adding services.
- 4 Click OK.

The target node will be added to the list of targets in the Connectivity Test Results window.

Continuously Poll the Services of the Target Nodes on a Segment Until the Test Is Disabled

For polling the services on the nodes of a segment, select the nodes on a segment with the list of services you want to test. Enable the test in the Monitor Tab Services window and view the results of the test in the Polling view.

If you do not disable the test, polling of the services continues after you close the console.

To define the services on the nodes for polling:

- 1 From ConsoleOne, right-click the node of a segment > click Properties > click the Monitor Services tab.
The List of Segment dialog box displays the different addresses of the same node on different segments if the node is connected to more than one segment. Click the node on the segment that you want to add.
- 2 Specify the details for the target nodes in the Monitor Services Tab window.

Refer to “[Adding Services for Monitoring](#)” on page 190 for more information about adding services.

3 Click OK.

Displaying Test Results Data

After defining the services for testing on the target node, you can view the results from the console.

Depending on the test you choose, the test results are displayed in the corresponding window.

If you choose to test the services on the node one time only, the test results will be displayed in the Ping Status window of the Ping window. This target will not be tested in the Connectivity Test Results window.

If you choose to continuously monitor the services, the test continues until you close the window. You can view the results in the Connectivity Test window.

If you choose to continuously poll the services until you disable the test, you can view the test results in the Polling view.

The following test data is available when you monitor the services on the target nodes:

Ping Target: Name or address (IP or IPX) of the network device for which services are being tested.

Service: Monitored services that are being tested on the target.

Port: Port number that the service uses.

Status of the Target: Up Status means that the service is available on the node and can be reached from the remote ping server. Down Status means that the service is down and cannot be reached from the server.

Round Trip Delay: Time interval (in milliseconds) between the instant the remote ping server sends the test packet to the target and the instant the response is received from the target.

Packets Sent: Number of packets sent from the remote ping server to the target node.

Packets Received: Number of packets received by the remote ping server from the target node.

Packets Lost: Number and percentage of packets lost during the testing of the target node.

Interval: Displays the test interval value, in seconds. This value determines the time duration between two successive tests on the target.

Timeout: Displays the timeout value, in milliseconds. This value determines the time duration that the remote ping server waits to receive the response from the target node.

To view the Connectivity Test Results window:

1 Click File > Action > Connectivity Test from the Console.

If you select one or more target nodes from the right pane of the console, the list of nodes that you want to test for connectivity will be shown in the Connectivity Test Results window.

To view the results of the polling:

1 From the console, click a segment > View > Polling.

NOTE: To delete a target node from the list, from the Polling view, click the target node > click Delete.

Changing the Test Options for a Node

You can modify the test options, such as the test interval and timeout options, that you set earlier on an individual node or on multiple nodes. To modify multiple nodes, click more than one node from the Connectivity Test Results window; the test options apply to all selected target nodes.

To view the Connectivity Test Results window:

- 1** Click the target row from the Connectivity Test Results window > click the Edit button.
- 2** Type values for the Ping Interval and Timeout.
- 3** Click OK.

If you want to roll back to the default setting, click Apply Defaults.

Adding Services for Monitoring

Monitoring Services lets you test services on the nodes. If you need to test any TCP-based service that is not listed in the default services list, you add the details of the service when you are adding the targets.

You specify the name of the service in the Add Service dialog box. Ensure that the service name you add is a unique name. Also, you must specify the port number for the service.

You can add the details of the service under the following circumstances:

- ◆ [“Continuously Monitor the Services on the Target Nodes” on page 188](#)
- ◆ [“Continuously Poll the Services of the Target Nodes on a Segment Until the Test Is Disabled” on page 188](#)

The services that you add are stored in a file on the server.

8

Understanding Traffic Analysis

ZENworks® for Servers (ZfS) provides traffic analysis tools that monitor network traffic, capture data, and collect key statistics of monitored segments nodes, and devices, allowing you to obtain, review, and analyze vital information to effectively troubleshoot and manage your LAN and keep your network operating at peak performance.

This section contains the following topics:

- ◆ [“Understanding Traffic Analysis” on page 191](#)
- ◆ [“Planning for Segment Monitoring” on page 204](#)
- ◆ [“Preparing to Analyze Network Traffic” on page 206](#)
- ◆ [“Analyzing Network Traffic” on page 208](#)
- ◆ [“Optimizing Traffic Analysis” on page 237](#)
- ◆ [“Understanding the Traffic Analysis Agents” on page 247](#)
- ◆ [“Using the Traffic Analysis Agent for NetWare” on page 249](#)
- ◆ [“Using the Traffic Analysis Agent for Windows NT/2000” on page 263](#)

Understanding Traffic Analysis

This section contains basic information to help you understand traffic analysis and describes the ZfS traffic analysis components.

- ◆ [“Traffic Analysis Components” on page 191](#)
- ◆ [“Communication Between Traffic Analysis Components” on page 192](#)
- ◆ [“Traffic Analysis Features” on page 193](#)
- ◆ [“Traffic Analysis Fundamentals” on page 194](#)

Traffic Analysis Components

The ZfS traffic analysis components include:

- ◆ [“Management Server” on page 191](#)
- ◆ [“Management Console” on page 192](#)
- ◆ [“Monitoring Agent Server” on page 192](#)

Management Server

The management server comes with the robust and highly scalable Sybase* Adaptive Server Anywhere that stores static information, such as the names and addresses of the nodes and devices in your network. The management server components include the NetExplorer™, management database, Consolidator, and Atlas Manager. NetExplorer discovers the objects in your network and stores them in the management server. The Consolidator takes the information about network objects discovered by NetExplorer and builds the management database. For details about the functionality of NetExplorer, see [“Understanding Network Discovery” on page 62](#).

The management database is comprised of the Common Information Model (CIM) schema that is used to establish the topology of the network. The CIM schema extension capabilities provide the ability to organize the information in the database and give this information the shape of a network map. The Atlas Manager obtains information from the management database and displays the network map on ConsoleOne.

Management Console

ConsoleOne[®], the Novell[®] directory-enabled, Java*-based network management and administration tool, is the management console component. ZfS snaps in to ConsoleOne and expands ConsoleOne's capabilities by adding menu options, property pages for existing Novell[™] objects, and ways to browse and organize network resources. ConsoleOne provides an intuitive, graphical user interface for ZfS traffic analysis. For details about the functionality of ConsoleOne, see [“Managing the Atlas” on page 99](#).

Monitoring Agent Server

Before you start analyzing segments or devices on your network, you need to ensure that they are monitored. To enable monitoring, make sure you have installed the network monitoring agent software either on the management server or on an independent server in your network. For more information, see [“Installing and Setting Up Management and Monitoring Services”](#) in the *Installation* guide. Network monitoring agents gather information or provide services that help you monitor your network.

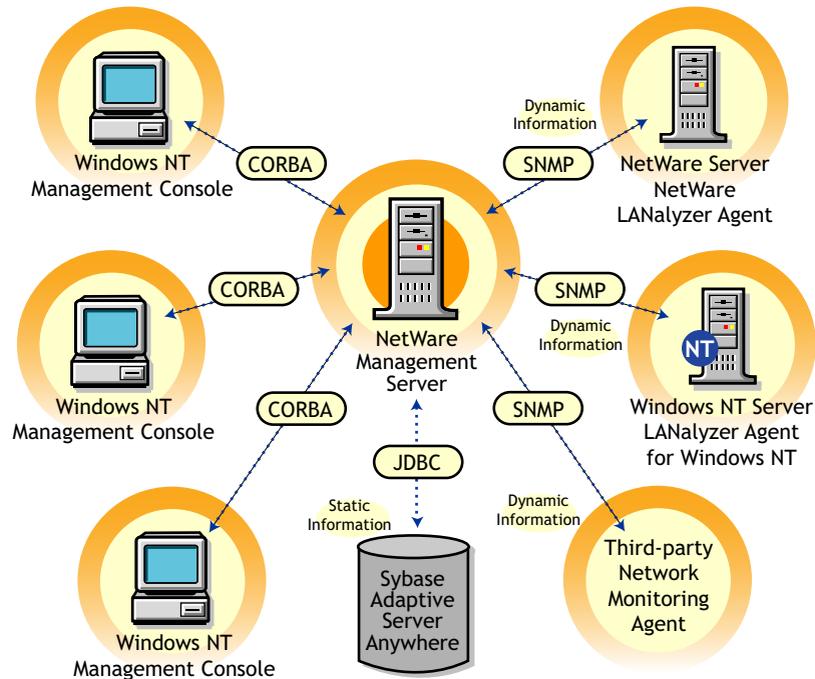
An agent program using parameters you have provided searches all or part of your network, gathers information you query, and presents it to you when you require it. You can use the information gathered by the agent to analyze the traffic on your network. The agent also warns you of problems, such as duplicate IP addresses, by sending an alert to ConsoleOne to help you solve problems before network performance is impacted. For details about managing alarms, see [“Managing the Alarm Management System” on page 109](#).

Network monitoring agents observe traffic and capture frames to build a database of network objects and information to help you detect network aberrations. With the network monitoring agent software installed on a server on each of your segments, you can use the traffic analysis tools to help you monitor the traffic on your network, identify the source of network problems, and maintain optimum performance. For details, see [“About Network Monitoring Agents” on page 194](#). The traffic analysis agents for NetWare[®] and Windows* NT*/2000 are part of ZfS that you can use to monitor Ethernet, FDDI, or token ring networks.

Communication Between Traffic Analysis Components

ConsoleOne communicates with the management server using common object request broker architecture (CORBA) to procure dynamic and static information about the nodes and devices in your network. When ConsoleOne requests static information from the management server, the management server communicates with the management database using Java Database Connectivity (JDBC), gathers the required static information from the database, and provides it to ConsoleOne. When ConsoleOne requests dynamic information from the management server, the management server communicates with the network monitoring agent using SNMP, gathers the required dynamic information, and provides it to ConsoleOne.

The following diagram illustrates this communication:



Traffic Analysis Features

The ZfS traffic analysis components provide the following features:

- ◆ “Analyze Traffic Generated by Segments” on page 193
- ◆ “Analyze Traffic Generated by Nodes Connected to Segments” on page 193
- ◆ “Capture Packets, Decode Captured Packets, and Display Captured Information” on page 194
- ◆ “Analyze Traffic Generated by Protocols” on page 194
- ◆ “Analyze Traffic Generated by Switches” on page 194

Analyze Traffic Generated by Segments

You can use the traffic analysis tools to collect current and historical segment statistics that can be displayed in real time, stored for later display, or transferred to a database, spreadsheet, or management reporting system. For details, see “Analyzing Traffic on Segments” on page 209.

Analyze Traffic Generated by Nodes Connected to Segments

The traffic analysis tools allow you to obtain statistical information about nodes on monitored Ethernet, FDDI, or token ring segments, and determine the top nodes on a segment. You can monitor the status of nodes in your network so that you are alerted when a node becomes inactive. You can also view alarms that are generated when preset threshold parameters are exceeded. Alarms that require immediate attention can be forwarded via e-mail to remote users. For details, see “Analyzing Traffic on Nodes Connected to a Segment” on page 216.

Capture Packets, Decode Captured Packets, and Display Captured Information

You can use the traffic analysis tools to capture packets between nodes on a monitored segment, and you can quickly define a capture filter based on which you want the packets to be captured. After packets are captured, protocols are decoded and displayed in color-coded summary, decode, and hex panes. The information obtained from the captured packets can be used to examine the traffic on the segment and to analyze it. By providing analysis capabilities and advanced protocol decodes, the traffic analysis tools allow you to identify network aberrations and resolve network performance problems. For details, see [“Capturing Packets” on page 223](#), [“Protocol Decodes Suite Supported by ZfS” on page 203](#), and [“Displaying Captured Packets” on page 226](#).

Analyze Traffic Generated by Protocols

You can use the traffic analysis tools to determine the distribution of protocols in the network, transport, and application layer of your network, and obtain statistical information of protocols discovered by the network monitoring agent. For details, see [“Analyzing Traffic Generated by Protocols in Your Network” on page 232](#).

Analyze Traffic Generated by Switches

You can analyze switch traffic by using the traffic analysis tools to determine port statistics of monitored switches. For details, see [“Analyzing Traffic on Switches” on page 235](#).

Traffic Analysis Fundamentals

ZfS provides tools to let you obtain statistical information about segments, nodes, and devices on your network. You can use this information to analyze and manage the performance of traffic on your network to help you keep the network operating smoothly. ZfS also provides tools to capture and decode packets between nodes. You can use the decoded information obtained from captured packets to analyze the traffic between nodes.

To be able to analyze the segments and nodes connected to a segment, you need to ensure that the segment is monitored by a network monitoring agent. You choose the agent based on the type of your network. The ZfS traffic analysis tools include the Traffic Analysis Agent for NetWare and Traffic Analysis Agent for Windows NT/2000, which you can use to monitor segments in your network. NetWare 5.x, the management server for ZfS, includes eDirectory, which is leveraged by ConsoleOne, to enable role-based administration.

The following sections provide information that will help you understand the ZfS traffic analysis functionality:

- ◆ [“About Network Monitoring Agents” on page 194](#)
- ◆ [“Role-Based Traffic Analysis Tasks” on page 202](#)
- ◆ [“Protocol Decodes Suite Supported by ZfS” on page 203](#)

About Network Monitoring Agents

Network monitoring agents provide the functionality to remotely monitor segments and devices on your network using SNMP. The agents collect and store statistical and trend information about nodes and devices on the network to provide real-time information about the status of your network. From your desktop, the agents let you troubleshoot and optimize Ethernet, FDDI, or token ring segments.

Based on the size and type of your network, you can use RMON, RMON Lite, RMON Plus, RMON2, or Bridge agents to monitor traffic. The following sections provide information to help you understand the functionality of agents:

- ◆ “Functionality of RMON Agents” on page 195
- ◆ “Functionality of RMON Lite Agents” on page 196
- ◆ “Functionality of RMON Plus Agents” on page 197
- ◆ “Functionality of RMON2 Agents” on page 198
- ◆ “Functionality of Bridge Agents” on page 200
- ◆ “Viewing the Summarized RMON Information” on page 201

Functionality of RMON Agents

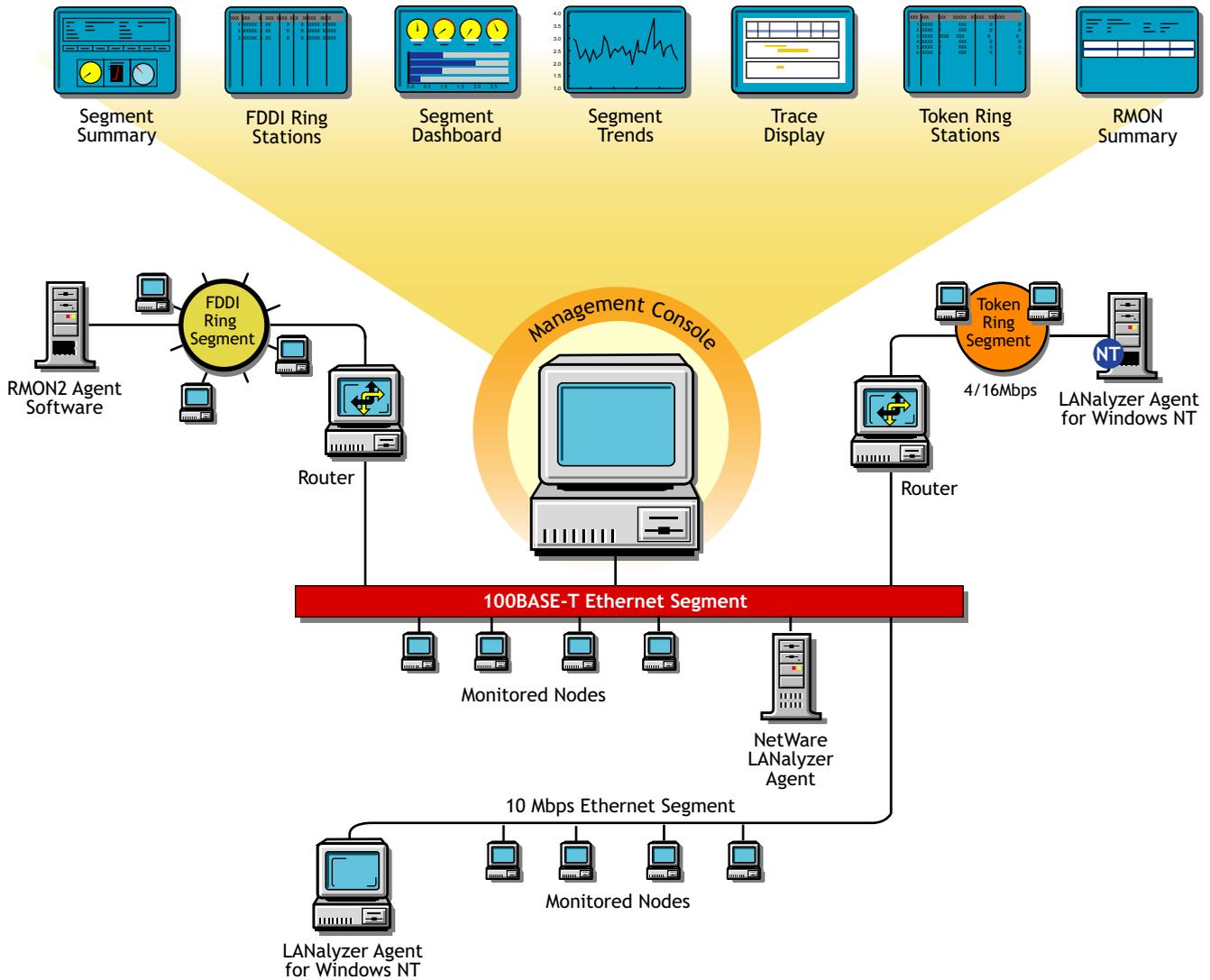
RMON agents use a standard monitoring specification that allows various nodes and console systems on your network to exchange network data. This data can be used by a network administrator to monitor, analyze, and troubleshoot a group of distributed LANs from a central site. RMON is specified as part of the MIB in [RFC 1757 \(http://www.isi.edu/in-notes/rfc1757.txt\)](http://www.isi.edu/in-notes/rfc1757.txt) as an extension of the SNMP.

RMON agents are ideally used for monitoring Ethernet, FDDI, or token ring segments.

RMON agents collect information in the following nine RMON groups of monitoring elements, each providing specific sets of data to meet network monitoring requirements. For details, see [RFC 1757 \(http://www.isi.edu/in-notes/rfc1757.txt\)](http://www.isi.edu/in-notes/rfc1757.txt).

RMON Group	Description
Statistics	Contains statistics measured by the agent for each monitored interface on the device.
History	Records periodic statistical samples from a network and stores them for later retrieval.
Alarm	Periodically takes statistical samples from variables in the agent and compares them with previously configured thresholds. If the monitored variable crosses a threshold, an event is generated.
Host	Contains statistics associated with each host discovered on the network.
HostTopN	Prepares tables that describe the hosts that top a list ordered by one of their statistics.
Matrix	Stores statistics for conversations between sets of two nodes. As the device detects a new conversation, it creates a new entry in its table.
Filters	Allows packets to be matched by a filter. These matched packets form a data stream that may be captured or generate events.
Packet Capture	Allows packets to be captured after they flow through a channel.
Events	Controls the generation and notification of events from the device.

The following figure illustrates the ZfS views that you can display when you use an RMON agent to monitor the nodes and devices on your network.



Functionality of RMON Lite Agents

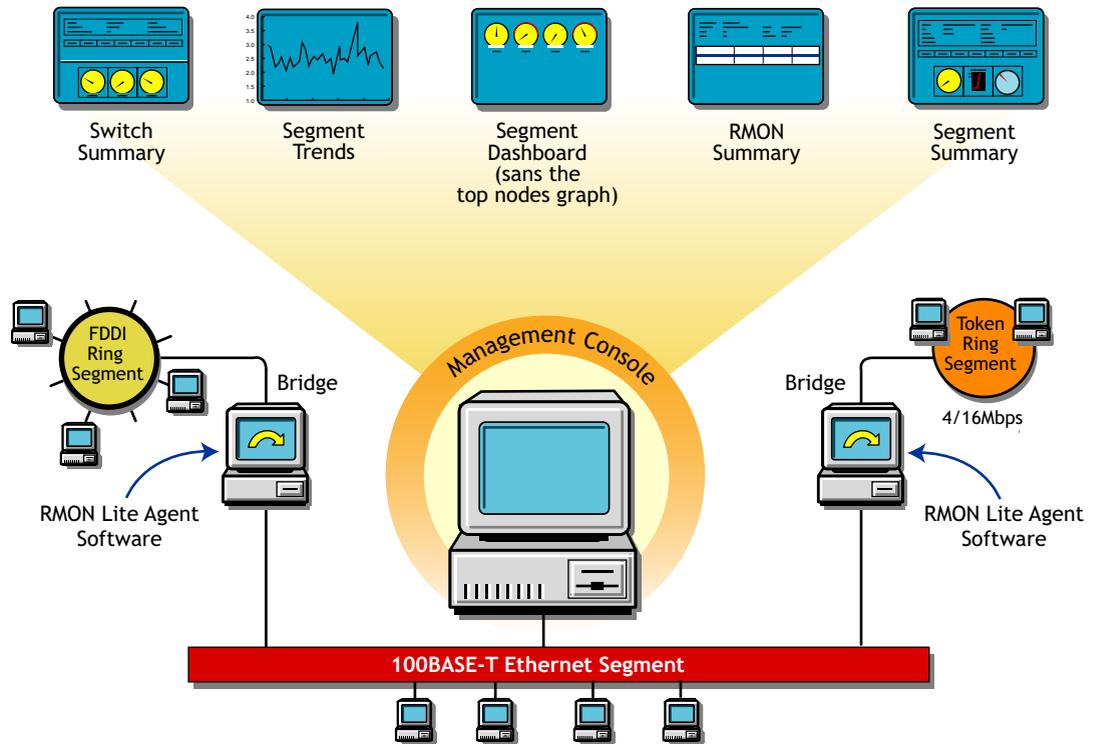
RMON Lite agents are ideally used for monitoring devices not dedicated for network management. For example, RMON Lite agents can be used to monitor a switch in your network.

RMON Lite agents support the following four RMON groups:

- ◆ Statistics
- ◆ History
- ◆ Alarm
- ◆ Event

Refer to the table in “[Functionality of RMON Agents](#)” on page 195 for a brief description of each group.

The following figure illustrates the ZfS views that you can display when you use an RMON Lite agent to monitor the nodes and devices on your network.



Functionality of RMON Plus Agents

RMON Plus agents are proprietary agents that extend the functionality of the RMON agent by providing data collected from the RMON groups, explained in [“Functionality of RMON Agents” on page 195](#), and the groups explained in the following table.

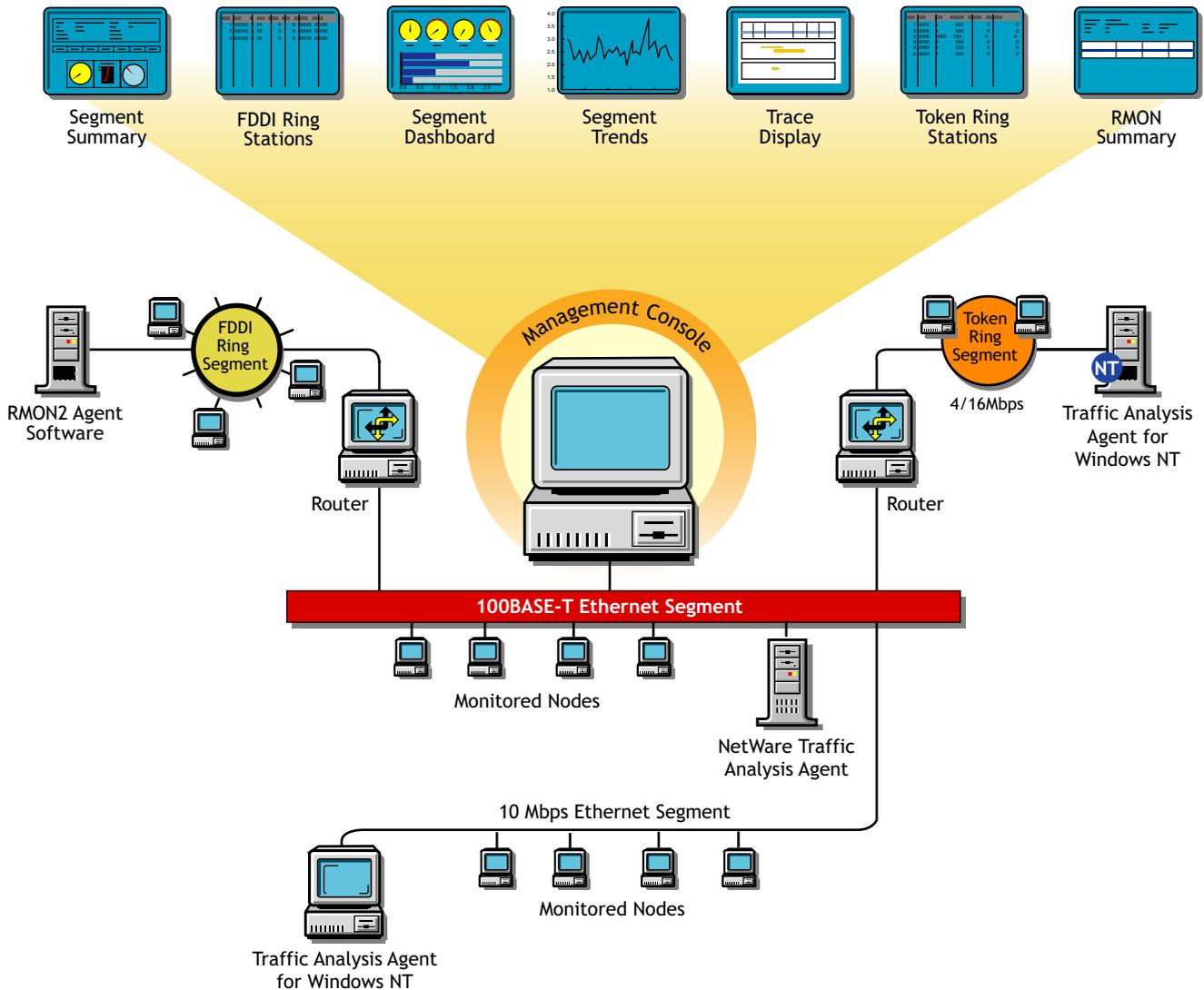
RMON Plus Group	Description
Buffer	Records the number of octets (excluding framing bits but including frame check sequence [FCS] octets and overhead) in packets which are captured in the buffer.
Admin	Collects information specific to the agent, such as the version number.
HostMonitor	Monitors a set of nodes for a particular host table and sets traps when a host becomes active or inactive.
DuplicateIP	Records and updates a list of packets arriving with duplicate IP addresses.
MacToIP	Stores records of the IP addresses associated with a host address for an individual host table.
BoardStatus	Records the status of each logical interface of the RMON agent.

RMON Plus agents are ideally used for monitoring Ethernet, FDDI, or token ring segments. Data from different media types can be collected based on the version of the RMON Plus agent that is used to monitor traffic on your network. Refer to the following table to determine the media type support based on the version of the RMON Plus agent.

RMON Plus Agent	Media Support
Traffic Analysis Agent for NetWare 1.1	Ethernet and token ring

RMON Plus Agent	Media Support
Traffic Analysis Agent for NetWare 1.21 or later	Ethernet, FDDI, or token ring
Traffic Analysis Agent (version 1.30) for Windows NT/2000	Ethernet, FDDI, or token ring

The following figure illustrates the ZfS views that you can display when you use an RMON Plus agent to monitor the nodes and devices on your network.



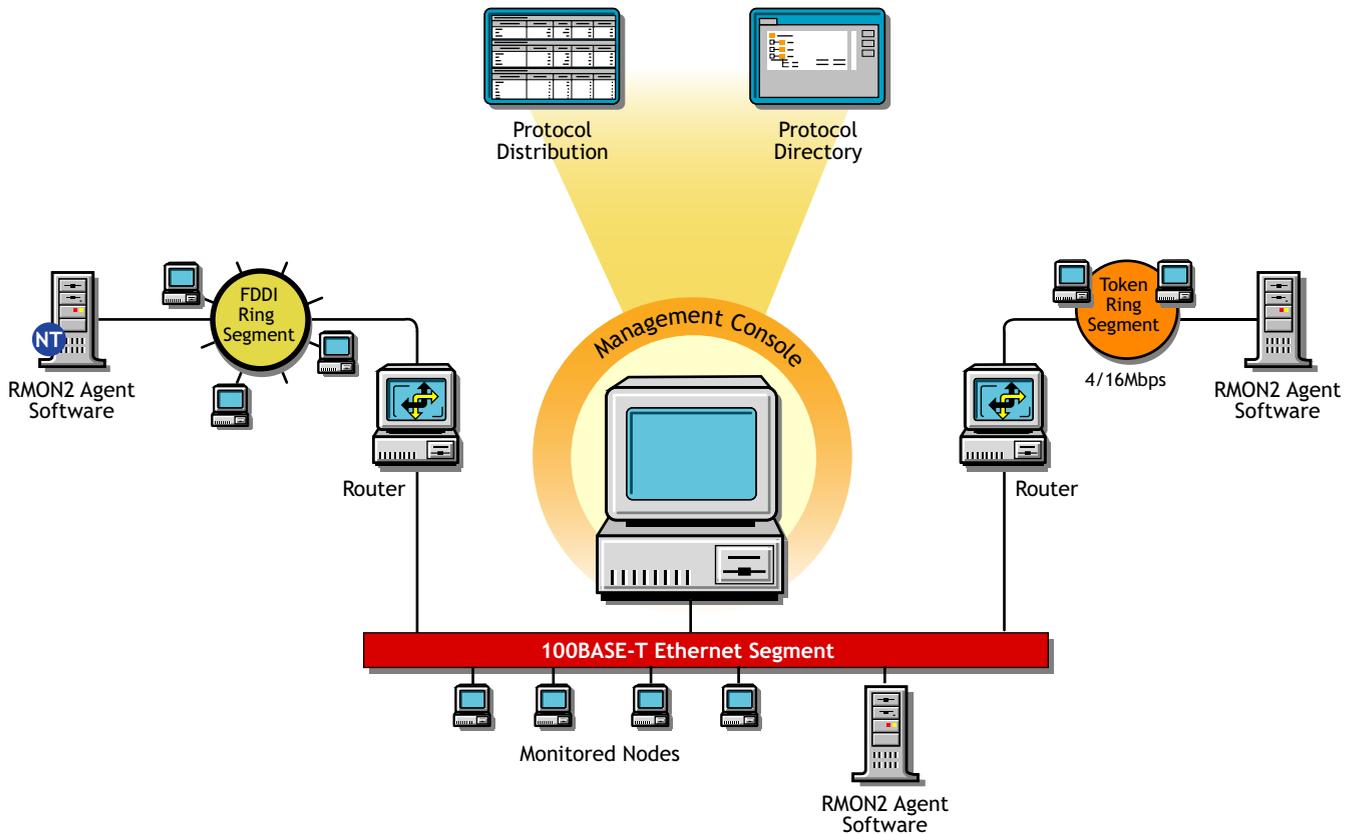
Functionality of RMON2 Agents

RMON agents can be used to collect data from nodes and devices in the physical and the data link layers and RMON2 agents can be used to collect data from nodes and devices in the network and application layers of your network. RMON2 agents can also determine network usage based on the protocol and application used by the nodes in your network. The following RMON2 groups make it possible to view traffic patterns above the data link layer. For details, see [RFC 2021 \(http://www.isi.edu/in-notes/rfc2021.txt\)](http://www.isi.edu/in-notes/rfc2021.txt).

RMON2 Group	Description
Protocol Directory	Provides a table of all identifiable protocols and their descriptions.
Protocol Distribution	Provides statistics for each protocol that the agent is configured to track.
Address Map	Maps a network layer address to the corresponding Media Access Control (MAC) address.
Network-Layer Host	Provides statistics for each host by network layer address.
Network-Layer Matrix	Provides statistics for each network conversation between pairs of network layer addresses.
Application-Layer Host	Provides statistics on traffic generated by each host for a specified application layer protocol. Traffic broken down by protocols can be recognized by the Protocol Directory group.
Application-Layer Matrix	Provides statistics on conversations between pairs of network layer addresses for a specified application layer protocol. Traffic broken down by protocols can be recognized by the Protocol Directory group.
User History	Enables the agent to save samples of RMON2 data for any MIB object at specified intervals.
Probe Configuration	Provides remote capability for configuring and querying agent parameters such as resets, software updates, IP address changes, and trap destinations.
RMON Conformance	Provides information to management software regarding the status of support for the groups.

IMPORTANT: The Console supports only the Protocol Directory and Protocol Distribution groups.

The following figure illustrates the ZfS views that you can display when you use an RMON2 agent to monitor the nodes and devices on your network.



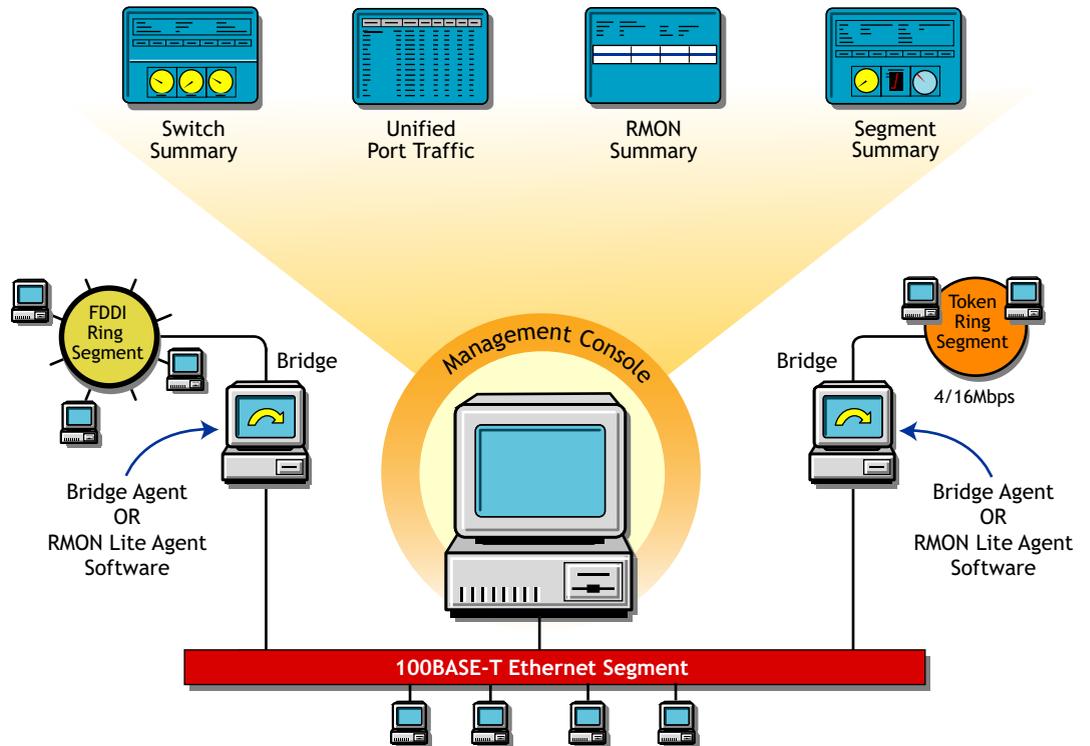
Functionality of Bridge Agents

Bridges are used to connect LAN segments below the network layer. A bridge connects two or more physical networks, forwarding packets between networks based on the information in the data link header.

Bridge agents collect information in the following five Bridge groups. You can use this information to monitor switched networks. For details, see RFC 1493 (<http://www.isi.edu/in-notes/rfc1493.txt>).

Group	Description
Base	Stores information about objects that are applicable to all types of bridges.
Spanning Tree Protocol	Stores information regarding the status of the bridge with respect to the Spanning Tree Protocol.
Source Route Bridging	Provides information that describes the status of the device with respect to source route bridging.
Transparent Bridging	Provides information that describes the entity's state with respect to transparent bridging.
Static	Collects information that describes the entity's state with respect to destination address filtering.

The following figure illustrates the ZfS views that you can display when you use a Bridge agent to monitor the nodes and devices on your network.



Viewing the Summarized RMON Information

The RMON Summary view provides brief information about RMON service on a selected node. It displays static information about the RMON agent and details of the resources requested by the user from the agent. The resource requests that are displayed in the RMON Summary view are Packet Capture and Host TopN requests.

To view the summarized RMON information:

- 1** Click RMON under Services within a node.
- 2** Click View > RMON Summary.

The following table describes the static information displayed in the RMON Summary view.

Statistic	Explanation
Agent Name	Name of the RMON agent monitoring the selected segment
IP Address	IP address of the node on which the RMON agent is installed
IPX™ Address	Internetwork Packet Exchange™ (IPX) address of the node on which the RMON agent is installed
Number of Interfaces	Number of logical interfaces for the management server on which the RMON agent is installed
Version	Version number of the RMON Plus agent
Type of RMON Service	Type of the RMON agent: RMON, RMON Plus, or RMON2
Status of the Agent	Status of the RMON agent

The RMON Summary view displays the resource information described in the following table.

Statistic	Explanation
Resource Name	Type of resource requested: <ul style="list-style-type: none">◆ Packet Capture◆ Host TopN
Owner	Owner string corresponding to the control entry of the row
Index	Channel, Filter, or Buffer control indexes for the Packet Capture resource and the Control index for the Host TopN resource

To delete a resource:

- 1** Select a row from the Resource table.
- 2** Click Delete.

When you delete a resource, the entry on the agent corresponding to the selected row is deleted.

Role-Based Traffic Analysis Tasks

ZfS lets you perform the following traffic monitoring tasks based on your role:

- ◆ Add nodes to be monitored for inactivity.
For details, see [“Monitoring Nodes for Inactivity” on page 221](#).
- ◆ Add protocols to the protocol directory tree.
For details, see [“Displaying a List of Protocols Used in Your Network” on page 232](#).
- ◆ Capture packets.
For details, see [“Capturing Packets” on page 223](#).
- ◆ Disable nodes from being monitored for inactivity.
For details, see [“Monitoring Nodes for Inactivity” on page 221](#).
- ◆ Delete protocols from the protocol directory tree.
For details, see [“Displaying a List of Protocols Used in Your Network” on page 232](#).
- ◆ Free agent resources.
For details, see [“Viewing the Summarized RMON Information” on page 201](#).
- ◆ Set segment alarms.
For details, see [“Configuring Alarm Options from the Set Alarm Dialog Box” on page 244](#).
- ◆ View conversations.
For details, see [“Viewing Conversations \(Traffic\) Between Nodes” on page 220](#).
- ◆ View Traffic Analysis Agents.
For details, see [“Selecting the Preferred RMON Agent” on page 206](#).
- ◆ View the protocol directory.
For details, see [“Determining the Distribution of Protocols in a Segment” on page 234](#).

- ◆ View the RMON summary.
For details, see [“Viewing the Summarized RMON Information” on page 201.](#)
- ◆ View segment alarms.
For details, see [“Viewing Alarm Statistics for a Segment” on page 214.](#)
- ◆ View the segment dashboard.
For details, see [“Determining the Performance of Individual Segments” on page 210.](#)
- ◆ View segments monitored for inactivity.
For details, see [“Monitoring Nodes for Inactivity” on page 221.](#)
- ◆ View segment protocol distribution.
For details, see [“Determining the Distribution of Protocols in a Segment” on page 234.](#)
- ◆ View segment stations.
For details, see [“Listing Statistics for Segments” on page 209.](#)
- ◆ View the segment summary.
For details, see [“Viewing the Summarized Segment Information” on page 215.](#)
- ◆ View segment trends.
For details, see [“Analyzing Traffic on Segments” on page 209.](#)
- ◆ View switch or port traffic.
For details, see [“Viewing Statistics for Ports in a Switch” on page 236.](#)
- ◆ View the switch summary.
For details, see [“Viewing the Summarized Switch Information” on page 236.](#)

For more information about role-based services, see [“Role-Based Administration” on page 32.](#)

Protocol Decodes Suite Supported by ZfS

ZfS decodes several protocol suites. Using ZfS, you can analyze and troubleshoot problems in the following protocol suites:

- ◆ NetWare Protocol Suite
- ◆ NetWork File System Protocol Suite
- ◆ Systems Network Architecture Protocol Suite
- ◆ AppleTalk* Protocol Suite
- ◆ TCP/IP Protocol Suite

You need to understand these protocols in order to set up packet capture and interpret the results in the Trace Display window. For more information about these protocol suites and decoding support, see [Appendix 10, “Protocol Decodes Suites Supported by ZfS,” on page 281](#)

ZfS also enables you to analyze and troubleshoot problems in the following media:

- ◆ Standard Ethernet
- ◆ IEEE 802.3
- ◆ Token Ring

- ◆ FDDI

Planning for Segment Monitoring

A baseline defines the typical activity of your network. Keeping a baseline document of activity on a segment lets you determine when the activity is atypical. Atypical activity might be caused by a problem or network growth. To create a baseline activity, you should gather statistical information when the network is functioning typically.

The following sections provide information about creating and using a baseline:

- ◆ [“Creating a Baseline of Typical Segment Activity” on page 204](#)
- ◆ [“Using the Baseline Document” on page 204](#)
- ◆ [“Segment Baseline Document Tips” on page 205](#)

Creating a Baseline of Typical Segment Activity

For segment statistics such as bandwidth utilization, you should create a trend graph that plots information over a period of time. Statistics sampling that gathers data over a short period of time can be misleading. If you have added one or more network components, it is useful to create another baseline against which you can compare future activity.

You can export the data you gather in ZfS into programs, such as spreadsheets, for further analysis and to maintain records over time.

Using the Baseline Document

You can use the baseline document for the following purposes:

- ◆ [“Using Baseline Documents to Set Alarm Thresholds Appropriately” on page 204](#)
- ◆ [“Using Baseline Documents to Track Network Growth and Its Effect on Performance” on page 205](#)
- ◆ [“Using Baseline Documents to Troubleshoot Atypical Segment Activity” on page 205](#)

Using Baseline Documents to Set Alarm Thresholds Appropriately

ZfS lets you set alarm thresholds for statistics on segments monitored by the network monitoring agent software, so that if the threshold is exceeded, you are notified at ConsoleOne. Setting alarm threshold values for statistics on a segment eliminates the need for you to constantly monitor segments for problems.

ZfS provides default values for thresholds of various alarms on Ethernet, FDDI, and token ring segments. Refer to the table in [“Configuring Alarm Options from the Set Alarm Dialog Box” on page 244](#) for a list of alarm statistics tracked by ZfS. By creating a baseline of activity on the segment, you can determine whether the default values are appropriate for segments in your network. For example, after tracking segment utilization, you would set an alarm threshold for bandwidth utilization at about 5% to 10% higher than typical utilization. You are then alerted if utilization is greater than usual for that segment.

IMPORTANT: If you want to use this alarm notification feature, you must enable segment alarms.

Using Baseline Documents to Track Network Growth and Its Effect on Performance

By comparing current network performance against the performance recorded in your baseline document, you can determine how performance is affected by network changes. This comparison also helps you plan for network growth and justify network upgrades and expansion. You can view graphs of real-time trends for various Ethernet, FDDI, and token ring statistics. If an RMON2 agent is installed on a segment, you can also view historical trends for those statistics over hourly, daily, weekly, monthly, and yearly periods. Refer to [“Analyzing Trend Data for a Segment” on page 211](#) for details about how to view a trend of segment performance. Refer to the table in [“Choosing Options to Display Stations on a Segment” on page 237](#) for a list of statistics based on which you can display a trend of segment performance.

Using Baseline Documents to Troubleshoot Atypical Segment Activity

By knowing what the typical network activity is, you can recognize atypical activity, which might help you isolate the cause of a problem.

Segment Baseline Document Tips

You should include the following key characteristics in each network baseline document:

- ◆ [“Bandwidth Utilization” on page 205](#)
- ◆ [“Packets Per Second” on page 205](#)
- ◆ [“Network Error Rates” on page 205](#)
- ◆ [“Kilobytes Per Second” on page 206](#)
- ◆ [“Most Active Servers on the Segment” on page 206](#)

Bandwidth Utilization

The bandwidth utilization statistic indicates the percentage of network bandwidth used. Bandwidth utilization is likely to be higher at certain times during the day (for example, when users log in to the network in the morning), week, or month. Tracking bandwidth utilization helps you balance traffic loads among network segments, servers, and routers for a more efficient network. This information also helps you determine the effect of network growth on performance. As new workstations and applications are added to a network, bandwidth utilization typically increases.

Packets Per Second

Monitoring the number of packets on the wire provides information about the traffic on the segment. By looking at the change in the packets per second after a user launches a new application, you can calculate what the increase in packets per second will be when all the users you expect to use the application start using it. Packets per second differs from utilization. Utilization is based on the number of kilobytes on the segment per second, but packets can range in size. Therefore, utilization can increase as a result of an increase in the size or number of packets. If the number of packets increases but utilization does not, it is likely that the number of small packets increased but the increase did not affect utilization.

Network Error Rates

By including error rates in your baseline, you can determine when error rates on the network are atypical. This is important because network errors can bring down the network. A higher error rate

can result from a hardware problem or network growth. If errors increase but utilization does not, there might be a problem with a component, for example a faulty network board or transceiver.

Kilobytes Per Second

Tracking kilobytes per second lets you determine the throughput of your network. From this information, you can determine the percentage of the total possible bandwidth that is in use. For Ethernet networks, the maximum possible utilization is 10 Mbps. For token ring networks, the maximum possible utilization is 4 or 16 Mbps (depending on the hardware).

Most Active Servers on the Segment

Keeping track of the top three servers on the network helps you distribute the load among them as you add new users and applications. See [“Viewing Statistics of the Top 20 Nodes” on page 217](#) for details about how to display a list of top nodes on a monitored segment. You should also monitor the number of Request Being Processed packets. A constantly increasing number of these packets indicates a server overload condition. You can monitor these packets by doing a packet capture and decode. See [“Capturing Packets” on page 223](#) and [“Displaying Captured Packets” on page 226](#) for details about how to capture and display decoded packets.

With the Segment Trends view, you can view many segment statistics and export that data into another application (such as a spreadsheet) for later analysis. The data is saved as a text file that stores statistical values of the trend you display. To export the trend data to a file, click the Export button in the toolbar of the Segment Trends view. For details, see [“Analyzing Trend Data for a Segment” on page 211](#).

You can view current utilization for a segment through the Segment Dashboard view. To access this view, select a segment, click View > click Segment Dashboard. For details, see [“Determining the Performance of Individual Segments” on page 210](#).

Preparing to Analyze Network Traffic

The ZfS software components include the Traffic Analysis Agent for NetWare and Traffic Analysis Agent for Windows NT/2000. You can install the network monitoring agent on the management server or on an independent NetWare or Windows NT/2000 server. The agent monitors the traffic on the segment it is connected to, gathers information about the nodes and devices on that segment, and makes this information available to the management server, which provides it to ConsoleOne. The agent also sends traps to the management server that are forwarded to ConsoleOne. The management server and the monitoring agent communicate using SNMP. ZfS provides default values for SNMP parameters.

The following sections provide information about specifying a preferred agent for monitoring traffic on the segment and changing the default SNMP settings:

- ◆ [“Selecting the Preferred RMON Agent” on page 206](#)
- ◆ [“Setting Up SNMP Parameters” on page 207](#)

Selecting the Preferred RMON Agent

If more than one remote monitor (RMON) agent exists on a selected segment, you can choose which agent is to monitor the nodes on the segment from the RMON Agent property page. This page displays a list of servers on which the RMON Agent is installed. The agent installed on the server that you choose from this list becomes the preferred agent. The preferred agent is the

primary agent that monitors the segment and sends information about segment activity to ConsoleOne.

To display the RMON Agent property page:

- 1 Select a segment from ConsoleOne.
- 2 Click File > Properties > the RMON Agent tab.

The following table describes the statistics displayed in the RMON Agent property page.

Statistic	Explanation
Preferred	Displays a check mark if the selected server is chosen to be the preferred RMON agent server.
Agent Name	Displays a list of all the servers on which the RMON agent is installed.
Version	Displays the version of the RMON agent installed on the server. The version is dynamically obtained. If ZFS cannot connect to the remote agent, or if a third-party agent is installed on the selected segment, this field is blank.
Status	Displays the status of the RMON agent on the segment.
MAC Address	Displays the physical Media Access Control (MAC) address of the node.
Interface Index	Displays the number of interface indexes in which each interface corresponds to a segment that the node can connect through the network board.
Available RMON Services	Displays the list of RMON services available from the selected agent: RMON, RMON Plus, or RMON2.

To choose an RMON agent as the preferred agent:

- 1 Choose a server or workstation name from the list of names displayed in the property page.
The server and workstation names displayed are those on which the RMON agent is installed.
- 2 Click Apply.

Setting Up SNMP Parameters

When you request dynamic information to be displayed in ConsoleOne, it seeks the information from the management server. The management server communicates with the network monitoring agent using SNMP, obtains the required information from the agent, and provides it to ConsoleOne. SNMP communications between the server and the agent are based on default SNMP settings provided by ZFS. You can change the default SNMP settings using the SNMP dialog box, which displays in ConsoleOne if an error occurs when the management server is communicating with the monitoring agent.

You can use the SNMP dialog box to specify the community strings and security settings for SNMP communication. You can change the default time-out value for the server to connect with the agent. If the default time-out value is exceeded before the server can communicate with the agent or if the community string of the server does not match that of the agent, the SNMP dialog box displays in ConsoleOne with the current settings. You can use the dialog box to change the

current time-out value, the community string, and other SNMP parameters. The changed values are saved in the ZfS database and will be applied for all subsequent traffic management sessions.

To change the SNMP settings for all monitoring agents in your network:

- 1 Right-click the ZfS domain from ConsoleOne > click Global SNMP Parameters.

To change the SNMP settings for a specific agent:

- 1 Right-click the node on which the agent is installed from ConsoleOne > click Properties > click SNMP Settings.

The following table describes the SNMP parameters displayed in the SNMP Settings property page.

Parameter	Explanation
Community String	Community string of the node requesting dynamic data from the agent
Timeout	Maximum duration the server should wait for a response from the agent
Retry	Number of times the server should try to connect with the agent
Secure Set	Encrypts the packet sent by the management server to the monitoring agent
Secure Get	Encrypts the packet sent by the monitoring agent to the management server

TIP: If the network monitoring agent is running on NetWare 4.x and your network is IPX enabled, use the SNMP dialog box to communicate with the agent using IPX. This will significantly improve the performance of ZfS traffic analysis components.

Analyzing Network Traffic

You can use ZfS to monitor your network and collect information such as a summary of real-time statistics to determine the performance of your network, or detailed real-time statistics to determine the performance of segments in your network.

Information about the activity of nodes and segments in your network is presented in views containing tables, dials, and graphs. You can use the information to perform various traffic management tasks such as establishing a baseline on your network to help you identify typical traffic loads and control network problems, and analyze real-time performance to help you balance traffic loads among network segments, servers, and routers. You can also collect node information to help you focus on specific entities that might be the source of problems.

The following sections provide detailed information about how you can use ZfS to manage your network monitoring activities:

- ◆ [“Analyzing Traffic on Segments” on page 209](#)
- ◆ [“Analyzing Traffic on Nodes Connected to a Segment” on page 216](#)
- ◆ [“Capturing Packets” on page 223](#)
- ◆ [“Displaying Captured Packets” on page 226](#)
- ◆ [“Analyzing Traffic Generated by Protocols in Your Network” on page 232](#)
- ◆ [“Analyzing Traffic on Switches” on page 235](#)

Analyzing Traffic on Segments

Monitoring the segments on your network helps you keep the network operating cost effectively, consistently, and smoothly. Based on the kind of information you want to obtain, you can choose the agent that will monitor the segments on your network. For details, see [“About Network Monitoring Agents” on page 194](#). The agent monitoring the segments will collect traffic data and provide real-time or historical information to you when you require it.

ZfS provides various views you can use to obtain statistical information about monitored segments. You can choose to view statistical information for all segments in your network or for individual segments. You can view a trend of segment performance and a list of alarms generated on a segment. The Segment Summary view provides a summary of segment performance.

The following sections provide information to help you analyze the performance of segments in your network:

- ◆ [“Listing Statistics for Segments” on page 209](#)
- ◆ [“Determining the Performance of Individual Segments” on page 210](#)
- ◆ [“Analyzing Trend Data for a Segment” on page 211](#)
- ◆ [“Viewing Alarm Statistics for a Segment” on page 214](#)
- ◆ [“Viewing the Summarized Segment Information” on page 215](#)

TIP: Servers running the remote monitor (RMON) agent can notify you when nodes you selected for monitoring become inactive. For details, see [“Monitoring Nodes for Inactivity” on page 221](#). Sometimes the RMON agent server must be taken off the network for maintenance. To prevent the segment from going unmonitored, you can choose a different RMON agent on the segment. For details, see [“Selecting the Preferred RMON Agent” on page 206](#).

Listing Statistics for Segments

The List Segments view displays a list of segments and statistical information for each segment on your network. Statistics are displayed in columns of the table in the view. The view displays a list of segments associated with the object or node you selected from ConsoleOne.

See [“Analyzing Traffic on Nodes Connected to a Segment” on page 216](#) for details about how to use ZfS to get information about nodes on individual segments.

To view statistical information of all segments:

- 1 Select an Area or a node from ConsoleOne.
- 2 Click View > List Segments.

If you select an Area, the List Segments view displays statistics for all segments found within that Area. If you select a node, statistics for all segments connected to that node will be displayed.

The following table describes the statistics displayed for each segment. The sampling interval for updating statistics on segments is 15 seconds.

TIP: Statistics of segments are displayed in the List Segments view only if the segments are monitored by a Traffic Analysis Agent for NetWare or Traffic Analysis Agent for Windows NT/2000.

Statistic	Explanation
Segment Name	Segment name or address.

Statistic	Explanation
Type	Physical segment type: Ethernet, FDDI, token ring, PPP, and unknown. Unknown indicates the segment whose physical segment type is other than the one listed.
Speed (Mbps)	The speed of the segment, as determined by the speed of the network board that attaches the RMON agent to the segment and factors such as the cable type of the segment. The value in this column appears only if you have at least one RMON agent connected to at least one server on your network.
Utilization%	Average percentage of the bandwidth currently used by all traffic on the segment.
Packets/s	Average number of packets per second currently transmitted on the segment.
KBytes/s	Average number of kilobytes per second currently transmitted on the segment.
Errors/s	Average number of errors per second currently appearing on the segment.
Message	Status of the RMON Agent on the segment. For details, see “Selecting the Preferred RMON Agent” on page 206 .

As ZfS polls segments, messages in the Messages column vary. These messages display the status of the preferred RMON agent on the segment.

The preferred RMON agent is the node you selected to send information about the segment to ConsoleOne. You can make this selection from the RMON Agent property page. For details, see [“Selecting the Preferred RMON Agent” on page 206](#).

You can modify the view to show fields; format columns; sort and group items; change the font of text fields; or display grid lines in the table view by selecting the required option from View > Settings. For details, see [Chapter 3, “Understanding Network Discovery and Atlas Management,” on page 61](#).

Determining the Performance of Individual Segments

ZfS provides real-time statistical information about the monitored segment on your network. This information is displayed in the Segment Dashboard view. The information displayed in this view is useful if you want to troubleshoot a segment.

The Segment Dashboard view displays four gauges that display the real-time statistics for a monitored segment. The lower portion of the view displays a bar graph of the top eight nodes, based on the value selected from the drop-down list. By default, it is based on packets out per second. See [“Viewing Statistics of the Top 20 Nodes” on page 217](#) for details about how to display a list of the most active nodes on a monitored segment.

You can configure the Segment Dashboard view to display the top eight nodes based on a different statistic. You can also choose to display or disable the top nodes graph. For details, see [“Choosing Options to Display Stations on a Segment” on page 237](#).

You can set alarm threshold values on segment alarms for packets per second, broadcasts per second, and utilization percentage statistics displayed in the Segment Dashboard view. For details, see [“Defining Alarm Thresholds for Statistics Displayed in the Segment Dashboard View” on page 211](#).

To view statistical information of an individual segment:

- 1 Select a segment from ConsoleOne.

2 Click View > Segment Dashboard.

The Segment Dashboard view displays four gauges that display real-time statistics for a monitored segment. The peak value is indicated by a line on each bar in the graph. The following table describes the statistics displayed in the Segment Dashboard view.

Statistic	Explanation
Packets/s	Number of packets per second currently transmitted on the segment
Utilization%	Percentage of maximum network capacity currently consumed by packet traffic on the segment
Error/s	Number of error packets per second currently transmitted on the segment
Broadcasts/s	Number of broadcast packets per second currently transmitted on the segment (a broadcast packet is sent to all addresses on the segment)

Statistics are updated every five seconds. The numeric value of each statistic is displayed in the gauge.

Defining Alarm Thresholds for Statistics Displayed in the Segment Dashboard View

To set alarm threshold values for statistics displayed in the Segment Dashboard view:

1 Click the black ring outlining the gauge.

2 Drag the ring to increase or decrease the default values.

As you drag the ring, the color of the ring changes to red.

3 Stop at the value you want to set as the threshold value for the statistic.

The color of the ring is displayed in red up to the selected threshold value.

If the statistic on the monitored segment exceeds the threshold value, the RMON agent sends a trap to the management server, which forwards it to ConsoleOne and an alarm is generated.

Viewing the Graph of the Top Nodes on a Monitored Segment

The lower portion of the Segment Dashboard view displays a bar graph of the top eight nodes on a monitored segment. The default statistic on which the graph is based is packets out per second. You can change the statistic on which the graph is based. For details, see [“Choosing the Statistic Based on Which Top Nodes Graph Is Displayed” on page 242](#). You can also choose to display or disable the top nodes graph. For details, see [“Choosing Options to Display the Top Nodes Graph” on page 241](#).

Statistics for the graph are updated every five seconds. Every 60 seconds, the graph is re-sorted and the new top nodes are displayed. At this point, new nodes might be added and existing nodes might be discarded from the list.

Analyzing Trend Data for a Segment

ZfS allows you to determine trends of traffic patterns on the monitored segment. You can view the trend of segment performance from the Segment Trends view. You can use trend information to create a baseline of typical activity on segments. Having a baseline helps you set appropriate thresholds for segment alarms and plan maintenance activities and backups. Additionally, if problems occur on the segment, you can compare the typical traffic level against the atypical traffic

level to help you discover the cause of the problem. For details, see [“Creating a Baseline of Typical Segment Activity” on page 204](#).

The following topics will help you analyze trend data:

- ◆ [“Understanding the Trend Display” on page 212](#)
- ◆ [“Viewing Trend Statistics” on page 212](#)

Understanding the Trend Display

Segment trend data is displayed depending on the type and settings of the RMON agent monitoring the selected segment.

- ◆ If RMON Plus is the segment’s preferred RMON agent, you can view current trends gathered every 30 seconds over the last hour and historical trends displayed over hourly, daily, weekly, monthly, or yearly periods.
IMPORTANT: If an RMON agent is installed on more than one node on a segment, the node you select in the RMON Agent property page as the node to send information about the segment to ConsoleOne is the preferred RMON agent server. For more details, see [“Selecting the Preferred RMON Agent” on page 206](#).
- ◆ If RMON Plus is not selected as the preferred RMON agent for the segment, you can view only the current trends for the selected segment. Current trends are gathered every 30 seconds over the last hour. Select an RMON Plus agent as the preferred RMON agent for the segment to be able to view historical trends.
- ◆ If the preferred RMON agent is Traffic Analysis Agent for NetWare version earlier than 1.30, you can view current trends gathered over the past hour and trends for the past day.
- ◆ Real-time trends will not be displayed if memory usage is excessive or if configuration settings in the RMON agent are unacceptable.
- ◆ If the RMON agent is down or is experiencing problems, the trend for a monitored segment will be displayed as a broken graph.
- ◆ If the preferred RMON agent is a Novell Traffic Analysis Agent (version 1.30 or greater) or a third-party agent that implements the token ring Extensions to the Remote Network Monitoring MIB (RFC 1513), the segment bandwidth utilization graph displays slightly lower values than the actual utilization in the trend for the token ring segment view. This is because the MAC layer statistics are not taken into consideration for the utilization calculation.

Viewing Trend Statistics

To view the trend statistics for a segment:

- 1 Select a segment from ConsoleOne.
- 2 Click View > Segment Trends.

Trend graphs are displayed for Ethernet, FDDI, and token ring segments. The default statistics, based on which graphs are displayed for the three types of segments, are as follows:

Segment Type	Default Statistic
Ethernet	Total packets, good packets, and error packets
FDDI	Total packets
Token ring	Total packets

The toolbar options let you change the time span of the trend you view, select statistics based on which you want the graph to be displayed, and export data to a file.

The following table describes the toolbar options in detail.

Option	Explanation	
Profile		<p>Displays the Profile dialog box, from which you can select a default profile. The default profile displays a trend with statistical information for total packets, good packets, and error packets on the monitored segment.</p> <p>If you choose not to use the profiles listed in the Select Profile list, you can select the required statistics from the Select Statistics list. You can save the selected statistics if you want to display the trend of a different segment based on the statistics you selected. The default profile will be enabled the next time you launch the Segment Trends view.</p>
Legend		<p>Shows what each color in the graph represents. The Legend can be resized.</p>
Stack		<p>Stacks the trends in a single graph representing all selected statistics, on a single vertical axis.</p>
Unstack		<p>Un-stacks the trends and displays the graph as a separate strip for each statistic.</p>
Horizontal Grid		<p>Displays horizontal grid lines in the graph area of the Segment Trend view.</p>
Vertical Grid		<p>Displays vertical grid lines in the graph area of the Segment Trends view.</p>
Scale To Fit		<p>Maximizes or minimizes the graph to fit the trend entirely in the graph area of the view.</p>
Export		<p>Copies the information in the Segment Trends view to a file. The file stores the statistical values displayed by the trend. You can save the data for later analysis.</p>

Option	Explanation
Time Scale drop-down list	 <ul style="list-style-type: none"> ♦ Real Time: Displays a current trend graph. The default sampling time for this graph is once every minute. This graph updates in real time. ♦ One Hour: Displays a historical graph of the selected trend with a time span of one hour. ♦ One Day: Displays a historical graph of the selected trend with a time span of one day. ♦ One Week: Displays a historical graph of the selected trend with a time span of one week. ♦ One Month: Displays a historical graph of the selected trend with a time span of one month. ♦ One Year: Displays a historical graph of the selected trend with a time span of one year. <p>Historical trends such as hourly, daily, weekly, monthly, and yearly trends are available only when Traffic Analysis Agent for NetWare version 1.1 or later is installed on the segment's preferred traffic analysis agent server.</p>

The File menu of the Segment Trends view can be used to print the statistical information of the current trend or to export the statistical information of a trend to a file and store the data in text format. You can later import the file into a spreadsheet for analysis.

You can view earlier or ensuing trends and change the size of the graph by using the options available in the graph area of the Segment Trends view, as shown in the following table.

Option	Description
Scale Up	 <p>Increments the Y-axis of the graph by half the current size with each click.</p>
Scale Down	 <p>Decrements the Y-axis of the graph by half the current size with each click.</p>
Previous	 <p>Displays the preceding graph based on the profile or statistics chosen. Enabled only when historical trends are displayed.</p>
Next	 <p>Displays the subsequent graph. Enabled only when historical trends are displayed.</p>

Viewing Alarm Statistics for a Segment

ZfS tracks alarm statistics for segments. Alarms are generated when threshold values for statistics on a segment are exceeded. You can view a list of all the alarms for the monitored segment in the Segment Alarms property page.

To view alarm statistics for a segment:

- 1 Select a segment from ConsoleOne.

2 Click File > Properties > the Segment Alarms tab.

ZfS provides default threshold values for various segment alarms. You can enable or disable the default values for a monitored segment. If you choose not to use the default values, you can set the threshold value using the Set Alarm dialog box. See [“Configuring Alarm Options from the Set Alarm Dialog Box” on page 244](#) for details about how to set segment alarms.

If a segment does not have an RMON agent connected to it, an error message is displayed.

Viewing the Summarized Segment Information

The Segment Summary view provides brief information about a monitored segment in your network. It displays static information about the monitored segment, whether the segment is monitored or not, and information about the alarms generated on the segment. At a glance, you can determine the utilization of network capacity by nodes on the monitored segment, view a trend based on packets transmitted by nodes on the segment, and see the distribution of protocols on the segment.

To view the summarized segment information:

- 1** Select a segment from ConsoleOne.
- 2** Click View > Segment Summary.

The following table describes the static information displayed in the Segment Summary view.

Statistic	Explanation
Name	Name of the segment
Type	Media type of the segment: Ethernet, FDDI, or token ring
IP Address	IP addresses of the segment
IPX Address	IPX address of the segment
Primary Agent	Name of the preferred agent monitoring the nodes and traffic on the segment
Agent Status	Status of the preferred agent monitoring the nodes and traffic on the segment
Nodes	Number of nodes on the segment
IP Nodes	Number of nodes on the segment that have an IP address
IPX Nodes	Number of nodes on the segment that have an IPX address
Servers	Number of NetWare servers on the segments
Workstations/Others	Number of nodes on the selected segment that are not NetWare servers
Network Probes	Number of monitoring agents on the selected segment
Switches	Number of switches on the segment
Routers	Number of routers used to connect nodes and devices on the segment
Hubs	Number of hubs on the segment

The Segment Summary view displays information about alarms generated on a monitored segment, as described in the following table.

Statistic	Explanation
Severity	Severity level attributed to the trap.
From	Network address of the device that sent the alarm to the alarm management system.
Summary	Summary of the event, often including the name or address of the object affected by the alarm.
Owner	Segment or device affected by the alarm.
Received Time	Date and time when the alarm management system received the alarm.
Type	Generic description of the alarm, for example, Volume out of disk space.
Category	Displays the category of the alarm based on the MIB that defines the trap-type objects. The category is directly related to the MIBs included in the management server MIB pool. For example, the category for NetWare servers is based on the NetWare Server Alarm MIB.

The Segment Summary view displays dynamic information about a monitored segment, as described in the following table.

Statistic	Explanation
Utilization%	Displays a dial representing the real-time values of the network capacity consumed by packet traffic on the segment.
Packets	Displays the trend based on packets transmitted on the segment. Displays real-time trends for segments monitored by RMON agents and daily trends for segments monitored by RMON Plus agents.
Protocol Distribution	Displays a pie chart representing the distribution of application layer protocols for which the agent monitoring the segment can collect data. Each slice represents a protocol suite. Click a slice to view the names of protocols. Enabled if the agent monitoring the selected segment is an RMON2 agent.

Analyzing Traffic on Nodes Connected to a Segment

ZfS provides various views you can use to obtain information about nodes connected to the monitored segments in your network.

The following sections provide information that will help you monitor the performance of nodes connected to the segments in your network:

- ◆ [“Viewing Statistics of the Top 20 Nodes” on page 217](#)
- ◆ [“Viewing Statistics of Nodes on an FDDI Segment” on page 218](#)
- ◆ [“Viewing Statistics of Nodes on a Token Ring Segment” on page 219](#)
- ◆ [“Viewing Conversations \(Traffic\) Between Nodes” on page 220](#)
- ◆ [“Monitoring Nodes for Inactivity” on page 221](#)

Viewing Statistics of the Top 20 Nodes

You can use ZfS to determine the statistics of the most active nodes on a segment for a wide range of performance statistics. This is useful if you want to discover which node is generating the most traffic based on a particular statistic. For example, you can find the heaviest source of broadcast traffic.

The Stations view displays a list of all nodes on a monitored segment. You can use this view to determine the top 20 nodes on a monitored segment. The view lists the top 20 stations sorted by packets out per second. You can choose a different statistic based on which you want the top 20 nodes to display. For details, see [“Choosing a Statistic Based on Which Top 20 Nodes Are Displayed” on page 238](#). If there are fewer than 20 top nodes, only the available number of top nodes are listed.

To view the statistics of the top 20 nodes on a segment:

- 1 Select a segment from ConsoleOne.
- 2 Click View > Stations.
- 3 From the Stations view, click View > Show Top N Stations.

The Stations view displays columns that provide statistical information for each station. The following table describes the statistics displayed in the Stations view.

Statistic	Explanation
MAC Address	Physical Media Access Control (MAC) address of a node
Node	Name of the node (or address, if the name is not in the database)
Util.%	Percentage of maximum network capacity consumed by packets sent by a node
Packets/s In	Packets per second received by a node
Packets/s Out	Packets per second transmitted by a node
Bytes/s In	Bytes per second received by a node
Bytes/s Out	Bytes per second transmitted by a node
Errors/s	Errors per second transmitted by a node
Broadcasts/s	Broadcast packets per second transmitted by a node
Multicasts/s	Multicast packets per second transmitted by a node (packets transmitted to a specific group of nodes)
Protocols	Types of protocols used by a node
First Transmit	Date and time a node first transmitted since the traffic analysis agent was started
Last Transmit	Date and time a node last transmitted since the traffic analysis agent was started

Stations statistics are updated periodically. Every 60 seconds, the table is resorted and new top nodes are displayed. At this point, new nodes might be added and existing nodes might be discarded from the list.

Viewing Statistics of Nodes on an FDDI Segment

ZfS lets you display data for nodes on monitored FDDI ring segments to help troubleshoot problems.

The FDDI Ring Stations view displays statistics for individual nodes on the monitored FDDI ring segment. The view lists the nodes on the segment and shows the order of each node on the ring and which node is the active monitor.

To view the statistics of nodes on an FDDI ring segment:

- 1 Select an FDDI ring segment from ConsoleOne.
- 2 Click View > FDDI Stations.

The statistics shown for each node are cumulative since the Traffic Analysis Agent for NetWare was last started and are updated every ten seconds as described in the following table:

Statistic	Explanation
Order	Relative position of the node on the FDDI ring from the traffic analysis agent.
Name	Name of the node or, if the name is not in the database, the physical (MAC) address of the node.
MAC Address	Physical (MAC) address of the node.
Status	Status of the node: <ul style="list-style-type: none">♦ On—The node is actively participating in a ring poll.♦ Off—The node is not participating in a ring poll.
Duration	Time elapsed since the node was On or Off.
UpStream Neighbor	MAC address of the node upstream to this station on the logical ring.
DownStream Neighbor	MAC address of the node downstream to this station on the logical ring.
Last Entered Time	Date and time the node last entered the ring.
Last Exit Time	Date and time the node last exited the ring.
SMT Request Type	The SMT request to which the node is responding. Indicates if the node was able to successfully respond to the request. In case of a failure, the response code indicates the reason.
SMT Response Type	The SMT response generated by the node on receiving an SMT request. If the node was unable to respond, the response code indicates the reason.
Request Denied	The cumulative total of request denied responses generated by the node. A request denied frame is generated when the responding node does not support the SMT version number of the requesting node, when a set fails, or when a request for synchronous bandwidth allocation by a node cannot be honored.
In CRC Error	Total number of cyclic redundancy check (CRC) line errors reported by this node.
Out CRC Error	Total number of CRC errors reported by the nearest active downstream neighbor of this station and detected by the probe.

Statistic	Explanation
Lost Frames	Total number of lost frame errors received on the network. A lost frame error indicates that the end delimiter of a frame was lost in the network.
In Beacons	Total number of beacon frames detected by the probe that named this station as its upstream neighbor.
Out Beacons	Total number of beacon frames sent by this station and detected by the probe.
Insertions	Number of times the probe detected this station inserting onto the ring.

Viewing Statistics of Nodes on a Token Ring Segment

The Token Ring Stations view displays statistics for individual nodes on the monitored token ring segment. The view lists the nodes on the segment and shows the order of each node on the ring and which node is the active monitor.

To view the statistics of nodes on a token ring segment:

- 1** Select a token ring segment from ConsoleOne.
- 2** Click View > Token Ring Stations.

The view displays statistical information as described in the following table. Statistics are cumulative since the RMON agent was started and are updated every ten seconds.

Statistic	Explanation
Order	Relative position of the node on the token ring from the RMON agent.
Name	Name of the node or, if the name is not in the database, the physical (MAC) address of the node.
MAC Address	Physical (MAC) address of the node.
Status	Status of the node: <ul style="list-style-type: none"> ◆ On—The node is on the ring. ◆ Off—The node is off the ring. ◆ On (Monitor)—The node is on the ring and is the active monitor.
Duration	How long this node has been on or off.
Last Entered Time	Date and time the node last entered the ring.
Last Exit Time	Date and time the node last exited the ring.
Duplicate Address	Total number of duplicate address errors reported, generated when this node detects other nodes using its own address.
Soft Errors	Number of soft errors in packets transmitted by this node.
Inline Errors	The total number of line errors reported by this station in error reporting packets to the ring error monitor and detected by the probe.

Statistic	Explanation
Outline Errors	The total number of line errors reported in error reporting packets sent by the nearest active downstream neighbor of this station and detected by the probe.
Internal Errors	Number of internal errors this node has reported. Internal errors generally indicate a recoverable failure of a network adapter board.
In Burst Errors	The total number of burst errors reported to the Ring error monitor and detected by the probe.
Out Burst Errors	The total number of burst errors reported in error reporting packets sent by the nearest active downstream neighbor of this station and detected by the probe.
AC Errors	Number of times this node could not interpret the Address Recognized Indicator (ARI) and the Frame Copied Indicator (FCI) during the ring process.
Abort Errors	Number of times a node transmitted an abort sequence. Abort sequences are usually transmitted when a node detects an error in frames it is currently transmitting.
Lost Frame Errors	Number of times a node transmitted a frame but failed to receive it back in its entirety.
Congestion Errors	Number of times the node detected a frame addressed to its specific address but could not copy it (generally due to insufficient buffers).
Frame Copied Errors	Number of times a node detected a frame addressed to its specific address with either or both the ARI and FCI bits set to 1. (Indicates that another node is using its address.)
Frequency Errors	Number of times a node's internal clock differed from the ring clock.
Token Errors	Number of token errors. These occur when the token gets corrupted or when the Active Monitor does not see a new frame transmitted in the required amount of time. Only the Active Monitor can report this error.
In Beacon Errors	The total number of beacon frames sent by this station and detected by the probe.
Out Beacon Errors	Total number of beacon frames sent by this station and detected by the probe.
Insertions	Number of times the probe detected this station inserting onto the ring.
Last NAUN	The station that was last named by the probe as the next active upstream neighbor (NAUN).

Viewing Conversations (Traffic) Between Nodes

ZfS provides real-time data about all the network traffic between a selected node and one or more other nodes on a segment. This data can be viewed from the Conversations view. You can use the data displayed in this view to determine specific information about node communication. For example, it can show which nodes communicate with a router or server, determine the load on a server, or examine the traffic flowing to or from a node that is reporting difficulties.

To view conversations between nodes:

1 Select a node from ConsoleOne.

2 Click View > Conversations.

If the selected node is connected to more than one segment, the Select Segment dialog box displays.

2a Select the segment where the node you want to examine traffic is connected > click View > click Conversations.

The Conversations view lists the percentage of traffic that each destination node contributes to the load on the source node. However, due to sample skewing (samples not taking place at the same time) and rounding up of statistics, the numbers in the columns do not always add up to 100%.

The statistics displayed in the Conversations view are updated every 5 seconds. The following table describes the statistics displayed in the Conversations view.

Statistic	Explanation
Node	Name of the destination nodes with which the source node is communicating
% Pkt Load	Percentage of the packet load between a destination node and the source node
% Byte Load	Percentage of the byte load between a destination node and the source node
Pkts/s In	Packets per second received by a destination node from the source node
Pkts/s Out	Packets per second transmitted by a destination node to the source node
Bytes/s In	Bytes per second received by a destination node from the source node
Bytes/s Out	Bytes per second transmitted by a destination node to the source node
Pkts In	Number of packets received by a destination node from the source node since the view was opened
Pkts Out	Number of packets transmitted by a destination node to the source node since the view was opened
KBytes In	Total kilobytes received by a destination node from the source node since the view was opened
KBytes Out	Total kilobytes transmitted by a destination node to the source node since the view was opened
Protocols	Protocol packet types used by the destination node in this conversation
First Transmit	Date and time that the destination node first transmitted on the network since the traffic analysis agent was loaded
Last Transmit	Date and time that the destination node last transmitted since the traffic analysis agent was loaded
MAC Address	Physical (MAC) address of the destination node

Monitoring Nodes for Inactivity

For segments on which at least one Traffic Analysis Agent for NetWare version 1.0 or later is installed, you can specify the nodes on the segment you want to monitor so that you are alerted if they become inactive. You can do this using the Monitor Nodes for Inactivity view.

Monitoring nodes for inactivity has the following advantages:

- ◆ You can monitor any node on the segment, regardless of the protocol the node uses.
- ◆ This feature does not impact network traffic because the traffic analysis agent does not poll the nodes to obtain their status.

To view a list of nodes monitored for inactivity:

- 1 Select a segment from ConsoleOne.
- 2 Click View > Monitor Nodes for Inactivity.

Another way to monitor connectivity is to specify the target in the Ping window and test the status of the specified node. The Connectivity Test window displays statistics that enable you to determine the status of the specified target. For details, see [Chapter 7, “Monitoring Services,” on page 185](#).

By default, the poll interval for refreshing the Monitor Nodes for Inactivity view is zero seconds. You can configure the poll interval based on how often you want the view to be refreshed. For details, see [“Specifying the Poll Interval for Refreshing the Monitor Nodes for Inactivity View” on page 247](#). You can also change the duration for the agent to verify the node before declaring it inactive. For details, see [“Specifying the Duration for the Agent to Determine if a Node Is Inactive” on page 247](#).

IMPORTANT: You do not need to keep the Monitor Nodes for Inactivity view open or ConsoleOne for the nodes to be monitored because the RMON agent is doing the monitoring, not ConsoleOne. The Alarm Manager must be running to record an inactive node in the Alarm Report. If ConsoleOne is not running, check for alarms after you restart it.

To monitor a node for inactivity:

- 1 Right-click a node from ConsoleOne or from any view that displays a list of nodes > click Monitor Nodes for Inactivity > click Add.

To disable a node from being monitored for inactivity:

- 1 Right-click the node that is monitored for inactivity > click Monitor Nodes for Inactivity > click Delete.

IMPORTANT: After the addition of any inactive node, if the NIC card of the node is changed, you will be able to see the node in the Monitor Node for Inactivity view but will not be able to delete it because of the change of MAC address.

Statistics displayed in the Monitor Nodes for Inactivity view are described in the following table.

Statistic	Explanation
Name	Displays a list of nodes that are being monitored for inactivity
MAC Address	Displays the MAC address of the network interface
Status	Displays the status of a node as active or inactive

You can open the Monitor Nodes for Inactivity view to check the Status column any time ConsoleOne is running. To do this, complete the following steps:

- 1 Select a segment from ConsoleOne.
- 2 Click View > Monitor Nodes for Inactivity.

The Status column displays if the selected node is active or inactive.

Capturing Packets

ZfS provides packet capture and decoding tools that help you analyze your network activity and identify the source of network problems. Capturing and decoding packets can help you troubleshoot network problems by giving you detailed information about what is actually happening on a segment.

ConsoleOne can request packet capture on any monitored segment. Each RMON agent captures packets on the segment it monitors and stores information in its local buffer.

The following sections contain detailed information about capturing packets:

- ◆ [“Defining a Capture Filter” on page 223](#)
- ◆ [“Starting Packet Capture” on page 225](#)
- ◆ [“Creating Simultaneous Packet Capture” on page 225](#)
- ◆ [“Stopping Packet Capture” on page 225](#)
- ◆ [“Restarting a Stopped Packet Capture” on page 225](#)
- ◆ [“Saving and Viewing the Captured Packets” on page 226](#)

Defining a Capture Filter

ZfS provides a capture filter with default values you can use to capture packets on any monitored segment. You can modify the values by defining a filter. For example, if you want to capture only NetWare packets sent by a certain node, you can define a filter to capture only those packets. As a result, the buffer has more space to store your selected packets.

When you specify a capture filter, you are specifying the packets to capture (include) in the buffer on the RMON agent, not the packets to exclude. When you specify both a node and a protocol, packets must meet both criteria to be captured. If you select more than one protocol family, packets can meet either protocol criterion to be captured.

To define a capture filter:

- 1** Select a node or a segment from ConsoleOne.
- 2** Click File > Actions > Capture Packets.
- 3** Type a name in the Buffer Name text box, if you do not want to use the default name.
The buffer name helps you keep track of multiple captures on the same segment.
- 4** Type or select the source and destination nodes from the Stations box. You can also click the Find Node icon to select the node from the Find dialog box, an atlas component.
The Stations box displays a list of nodes on the segment from which the user can capture packets. You can select from Hardware, IP, or IPX stations.
If you choose ANY in both the source and destination node list, all packets sent by or received from any node are captured.
- 5** Select the direction of traffic flow between the nodes.

Click an arrow option from the drop-down list to specify the direction of the traffic flow. The available node and traffic flow directions are shown in the following table.

Node	Arrow	Node	Effect
node1	<==>	node2	Capture packets that node1 sends to node2 and packets that node2 sends to node1.
node1	<==>	ANY	Capture packets that node1 sends to any node and packets that node1 receives from any node. This is equivalent to ANY <==> node1.
ANY	<==>	ANY	Capture all packets sent by or received from any node.
node1	==>	node2	Capture packets that node1 sends to node2. This is equivalent to node2 <== node1.
node1	==>	ANY	Capture packets that node1 sends to any other node. This is equivalent to ANY <== node1.
node1	<==	node2	Capture packets that node2 sends to node1. This is equivalent to node2 ==> node1.
node1	<==	ANY	Capture packets that any node sends to node1. This is equivalent to ANY<== node1.

- 6** If you want to filter on protocols used, add the protocol suites you want to the Selected list box.

To add a protocol to the Selected list box, select it from the Available list box > click Add.

or

To delete a protocol from the Selected list box, select it > click Remove.

All protocols are selected by default when you first use ZfS. If no protocols are listed in the Selected list box, all protocols are captured.

See [“Protocol Decodes Suite Supported by ZfS” on page 203](#) for details about the protocol decoding support that ZfS provides.

- 7** Specify what kind of packets to capture on Ethernet, FDDI, or token ring segments.

The default statistics for the segments are listed in the following table.

Segment Type	Available Statistics	Default Statistics
Ethernet	Only good packets, only error packets, or both good and error packets.	Good packets and error packets
FDDI ring	All packets, LLC packets, MAC packets, or SMT packets.	All packets
Token ring	All packets, non-MAC packets, or MAC packets. MAC packets are used to manage the operation of the token ring.	All packets

- 8** Specify whether to stop packet capture or to overwrite the oldest packets in the buffer with newer ones when the buffer is full.

Continuing packet capture means that a stop criteria does not exist and new packets will overwrite those already captured. You will need to manually stop packet capture if you select to overwrite the oldest packets.

9 Specify a buffer size.

Select a buffer size from the drop-down list or specify the size you want. The default buffer size is 32 KB.

The RMON agent will attempt to provide the buffer size requested. If not enough space is available in server memory for a large buffer, the RMON agent cannot create the requested size.

10 Select a slice size.

A slice specifies the maximum number of bytes of each packet, counting from the packet header, to keep in the buffer. This helps maximize the number of packets you can store in your buffer space, as well as reduce the load on the RMON agent to process captured packets. If you want to decode protocol header information, you need only 100 to 150 bytes. The rest is typically data that you need only if you suspect a data corruption problem. However, on certain very large packets, slicing can cause incorrect decodes by truncating information.

Your capture filter is now set up. If you decide not to capture packets, click the Cancel button.

Starting Packet Capture

To start packet capture:

- 1** Define a capture filter. See [“Defining a Capture Filter” on page 223](#) for the procedure.
- 2** Click OK to apply the filter settings on the preferred RMON agent of the segment.
- 3** Click Start in the Capture Status dialog box.

When you start packet capture, the Start button in the Capture Status dialog box toggles to read Stop and the activity indicator reflects the capture buffer storage as it progresses. As packets that meet the filter criteria are captured, the capture buffer will begin to store the packet data, and a box below it will display the number of packets captured. The needle stops turning when the capture buffer is full.

Creating Simultaneous Packet Capture

You can create simultaneous packet captures by repeating the procedure you followed to start the first capture. This lets you set up and run captures with different capture criteria.

You can run a maximum of 20 packet captures with different capture criteria.

Stopping Packet Capture

When you set up a capture filter, you choose whether to stop packet capture when the capture buffer is full or to continue to capture packets but overwrite the oldest packets in the buffer.

By default, the packet capture will stop when the capture buffer is full. If you select to overwrite when the buffer is full, you must stop packet capture manually.

To stop packet capture manually, click the Close button in the Capture Status dialog box.

IMPORTANT: If you restart packet capture from the Packet Capture Setup window, the existing buffer is deleted and refreshed.

Restarting a Stopped Packet Capture

When the Packet Capture Setup window is open, you can start and stop capturing packets using the Start/Stop toggle button in the Capture Status dialog box. If ZfS is capturing packets, the button

is labeled Stop; if it is not capturing packets, the button is labeled Restart. The RMON agent buffer is cleared when you restart.

Saving and Viewing the Captured Packets

You can save captured packets to a file and view as many files as you want, either while you are viewing a capture buffer or independently.

To view the saved packet capture files:

- 1 Click Tools > View Packet File.

The File Open dialog box is displayed.

- 2 Browse and select the packet capture file.

The .TR1 file extension will be appended automatically.

Displaying Captured Packets

You can display and view decoded packets stored in the capture buffer from the Trace Display window by clicking the View button in the Capture Status dialog box. If you display this window while packets are being captured, capture automatically stops.

ZfS retrieves packet data from the RMON agent only as necessary for ConsoleOne to decode and display the packets as you view them. This minimizes the amount of packet data transferred between the RMON agent and ZfS. If you prefer not to display all the packets you captured, you can create a display filter to display only a defined group of captured packets. For details, see [“Defining the Display Filter” on page 229](#).

The following sections provide information on how you can view captured packets and perform trace display operations:

- ◆ [“Viewing Captured Packets” on page 226](#)
- ◆ [“Filtering Packets for Display” on page 228](#)
- ◆ [“Defining the Display Filter” on page 229](#)
- ◆ [“Selecting and Decoding a Different Packet” on page 230](#)
- ◆ [“Highlighting Protocol Fields and Hexadecimal Bytes” on page 231](#)
- ◆ [“Saving Packet Files” on page 231](#)
- ◆ [“Opening Packet Files” on page 232](#)
- ◆ [“Printing Packets” on page 232](#)

ZfS provides default settings based on which captured packets are displayed in the Trace Display window. To change the default values provided for displaying captured packet, see [“Choosing Options to Display a Captured Packet” on page 243](#).

Viewing Captured Packets

You can use the Trace Display view to view the decoded packet capture information, the packet data in hexadecimal format, and a summary of the captured packets:

To view a captured packet:

- 1 Select a node or a segment from ConsoleOne.

- 2 Click File > Actions > Capture Packet.
- 3 Capture packets using the capture filter of your choice. See [“Defining a Capture Filter” on page 223](#) for details.
- 4 Click the View button in the Capture Status dialog box.

The Trace Display window contains three panes that display captured and decoded packets, as described in the following sections:

- ◆ [“Viewing the Packet Decode” on page 227](#)
- ◆ [“Viewing Packet Data in Hexadecimal Format” on page 227](#)
- ◆ [“Viewing a Summary of Captured Packets” on page 227](#)

When you view packets initially, the first packet in the Summary pane is highlighted and selected. The contents of that packet are displayed in the Decode pane. If you select a different packet in the Summary pane, it is highlighted and the Decode pane displays its decoded contents.

You can change the size of the Trace Display panes by dragging the divider between windows.

Viewing the Packet Decode

The Decode pane displays detailed information about the contents of a selected packet. The packet contents are interpreted (decoded) and displayed by protocol fields.

By default, the Decode pane displays fully decoded packet data. You can configure the Trace Display window to display the decoded packets either as full protocol decodes or by one line per protocol layer. See [“Choosing Options to Display a Captured Packet” on page 243](#) for details about how to change the default settings.

Viewing Packet Data in Hexadecimal Format

The Hexadecimal pane shows uninterpreted packet data in hexadecimal format. The ASCII or EBCDIC portion of the Hexadecimal pane (to the right) displays a dot for every hexadecimal byte that has no ASCII or EBCDIC equivalent.

The first column in the pane indicates the offset in hexadecimal bytes. The offset is the number of bytes counting from the beginning of the header. For example, the first three lines have the following offset:

- ◆ Hexadecimal 0—indicates zero offset
- ◆ Hexadecimal 10—indicates decimal 16 offset (16 bytes precede this)
- ◆ Hexadecimal 20—indicates decimal 32 offset (32 bytes precede this)

Regardless of whether you choose to display one-line decoded or fully decoded packets in the Decode pane, entire packets are displayed in the Hexadecimal pane. The Hexadecimal pane and the highlighting tool are especially helpful with the full-decode display when you are trying to associate protocol fields with specific bytes in a packet. For details, see [“Highlighting Protocol Fields and Hexadecimal Bytes” on page 231](#).

Viewing a Summary of Captured Packets

The Summary pane gives you an overview of the conversation between the source and the destination nodes. You can select a packet in this pane for further decoding and display in the other panes. You can scroll the pane horizontally, and you can change the size and position of the columns in the pane.

Statistical information about the captured packets displayed by the Summary pane is described in the following table:

Statistic	Explanation
No.	Numbers the packets in order of arrival at the traffic analysis agent.
Source	IP address, IPX address, or the physical (MAC) address of the node that sent the packet. Names are stored in the database. If no name is found in the database, the MAC address is displayed.
Destination	Node to which the packet was sent. The node is displayed as the IP address, IPX address, or the physical (MAC) address of the node.
Layer	Abbreviation of the highest protocol layer in the packet. It might display NCP for NetWare Core Protocol™ (NCP™) software, ether for the Ethernet data link layer, RTMP for the AppleTalk Routing Table Maintenance Protocol layer, or 802.2 for the IEEE 802.2 Logical Link Control layer. If you choose the full decode option, the Decode pane displays the full name of the protocol layer and all its fields. The Hexadecimal pane shows the entire packet.
Summary	Brief description of the contents of the highest protocol layer.
Error	Type of errors, if any, in the packet. This column is displayed only for Ethernet media.
Size	Number of bytes in the packet. Packet size always excludes the packet preamble and the CRC.
Absolute Time	Clock time on your computer when the packet arrived.
Interpacket Time	Time elapsed from the end of the preceding packet to the end of the current packet.
Relative Time	Time that elapsed since the arrival of the first packet still in the buffer.

Filtering Packets for Display

After you have captured packets, you can apply a display filter to the capture buffer and view only the packets that interest you. You can filter on node names or addresses, protocol families or protocol layers, or contents of a selected field. This is useful in situations when, after you have captured packets, you realize there is a problem with a specific workstation and you want to display only the packets it has sent or received.

Display filtering requires the transfer of a portion of every captured packet from the RMON agent to ConsoleOne. For large captures, this consumes time and network bandwidth. We recommend that you define very specific capture filters rather than filtering during display. However, subsequent filtering of the same capture does not result in additional data transfer from the traffic analysis agent because the data is already transferred to ConsoleOne. Therefore, it is much quicker to filter the same packet capture a second time.

Display filters affect only the display; they do not change the capture buffer. All captured packets remain in the capture buffer and are available for viewing with a different display filter or without any display filter.

You can define a display filter in either of two ways:

- ◆ From the Trace Display window, click View > Filter.

The Display Filter dialog box is displayed. For details, see [“Defining the Display Filter” on page 229](#).

- ♦ Double-click a packet in the Summary pane or double-click a selected protocol layer or field in the Decode or Hexadecimal pane.

A filter is set based on what you selected. You can also modify the filter information as needed. For details, see [“Point-and-Click Filtering” on page 230](#).

Defining the Display Filter

Capture packets using the capture filter of your choice. See [“Defining a Capture Filter” on page 223](#) for details. To define a display filter:

- 1** Select a segment from ConsoleOne.
- 2** Click File > Actions > Packet Capture.
- 3** Click the View button in the Capture Status dialog box.
- 4** With the Trace Display window displayed and active, click View > Filter.
- 5** Select the nodes from the drop-down lists. You can select from IP, IPX or MAC address.
Alternatively, you can enter a node name or address in place of ANY in either or both of the drop-down list boxes.
- 6** Select the direction of the traffic flow from the arrow options available in the drop-down list.
- 7** To display all the packets of a specific protocol layer:
 - 7a** Double-click a protocol suite name from the list of protocols to display a list of all the protocols in the suite.
 - 7b** Scroll through the list to find the protocol you want.
 - 7c** Select the protocol.
- 8** To display all the packets that have the same contents in a specific field:
 - 8a** Enter the offset in hexadecimal bytes.
You can count the offset in the Hexadecimal pane when the packet is decoded, using the offset column for guidance. See [“Viewing Packet Data in Hexadecimal Format” on page 227](#) for details.
 - 8b** Specify whether the offset is counted from the beginning of the packet or from the beginning of a protocol layer.
If you choose the protocol layer option, you must select a specific protocol in the Protocol box.
 - 8c** Enter the data that you want to include in the filter.
 - 8d** Specify the format in which you want the data to be displayed. Select from hexadecimal, ASCII, or EBCDIC format options.
You can also fill in the values using point-and-click filtering. See [“Point-and-Click Filtering” on page 230](#).
- 9** Click OK.

The dialog box closes and ZfS begins to select the required packets from the capture buffer.

If you have a large capture buffer, ZfS displays the initial packets that pass the filter. ZfS continues to filter in the background while you examine these packets.

The Summary pane shows the list of filtered packets that met the criteria in the display filter. You can view and decode them as described earlier in this section.

Point-and-Click Filtering

You can define a display filter using the point-and-click method by double-clicking a field in the Trace Display window.

To define a display filter using the point-and-click method:

- 1 To display only packets in one conversation (for example, between a node and a server), double-click a packet in that conversation in the Summary pane.

The Display Filter dialog box displays the source and destination of the selected packet. You can also modify the addresses, if needed. For example, you can change the destination address to ANY, the broadcast address, or a specific node address.

or

To display all the packets containing a specific protocol layer, double-click the protocol line in the Decode pane.

The Display Filter dialog box displays the protocol you selected.

or

To display all packets with the same contents as a specific field, double-click the field in the Decode pane.

The Display Filter dialog box displays the field, data, and type of data for the selected field.

or

To display all packets with the same content as a specific offset, click the field in the Hexadecimal pane.

The Display Filter dialog box displays the offset and the type of data for the selected field.

- 2 Click OK.

The dialog box closes and ZfS begins to select the packets from the capture buffer.

The Summary pane displays the list of packets that met the display filter criteria.

Selecting and Decoding a Different Packet

To select a different packet for decoding:

- 1 Select View > Go To.

You can also use the arrow keys on your keyboard to highlight a different packet.

- 2 Enter the packet number.

If the packet number specified is more than the total number of captured packets, an error message displays. If a display filter is set and the specified packet number has not passed the filter, then a packet closest to the specified packet is displayed.

Packets are retrieved from the RMON agent as you select their headers in the Summary pane using the mouse or the arrow keys. Using the Go To dialog box avoids transferring unwanted packet data from the RMON agent. Similarly, scrolling the Summary pane with the scroll button retrieves only the packet header data when creating the decode summary, whereas using the arrow keys retrieves all packet data.

Highlighting Protocol Fields and Hexadecimal Bytes

ZfS provides a highlighting tool that helps you associate protocol fields and hexadecimal bytes. Highlighting can be a useful training tool for new network managers who want to learn about protocol decoding.

You can use this tool in the following ways:

- ◆ Highlight a protocol layer in the Decode pane.
All bytes are highlighted in the selected protocol layer of the Hexadecimal pane.
- ◆ Click a field in any of the protocol layers in the Decode pane.
Associated bytes are highlighted in the Hexadecimal pane.
- ◆ Click hexadecimal bytes in the Hexadecimal pane.
All hexadecimal and ASCII or EBCDIC bytes of this field in the Hexadecimal pane are highlighted, and the associated field is highlighted in the Decode pane.
- ◆ Click ASCII or EBCDIC text in the Hexadecimal pane.
All hexadecimal and ASCII or EBCDIC bytes that belong to the field are highlighted in the Hexadecimal pane, and the associated field is highlighted in the Decode pane.

Saving Packet Files

You can save captured packets to a file and open the file later to analyze or print. When you save packets to a file, ZfS creates a binary file with the name you specify. You might want to save packets to a file in the following situations:

- ◆ To transfer the packets to another system or to send them for analysis.
- ◆ To apply a display filter to decoded captured packets so you can view only the packets that interest you. After you apply the display filter, you can save the filtered packets to a file.
- ◆ To compare packets saved from your buffer with other packets. You can either save the other packets, or view them from the capture buffer. You can view only one active capture buffer at a time. However, after you have saved packets to a file, you can open as many files as you want, and simultaneously view a capture buffer, if desired.

Packet files are compatible with the Traffic Analysis Agent for Windows NT/2000 and earlier versions of ManageWise[®]. Hence, packets captured and saved using Traffic Analysis Agent for Windows NT/2000 can be viewed using ZfS.

To save captured packets to a file while viewing the capture buffer:

- 1** Click File > Save As.

The Save Filtered Packets or Save Unfiltered Packets dialog box is displayed, depending on whether you filtered your packets.

- 2** Enter the name in the Filename text box.

The .TR1 file extension is appended automatically.

- 3** Click OK.

IMPORTANT: Filter out the captured packets you want to save. (See ["Filtering Packets for Display" on page 228.](#)) When you save packets, you save only those that pass the display filter. If you did not filter the display, all packets are saved.

Opening Packet Files

To open a packet file:

- 1 From the main menu of ConsoleOne, click Tools > View Packet File.
- 2 Double-click the file you want to open.

Printing Packets

To print packets:

- 1 Open a Trace Display window, either by capturing packets or by opening a packet file.
- 2 Click File > Print.
- 3 Select the print options you want.

You can select the destination, format, and the packets you want to print.

- ◆ Choose whether to print to your default printer or to a file. If you choose a file, enter its name and specify whether the current packet data should overwrite the file or be appended to it.
 - ◆ Choose whether you want a summary of the packet information, only the hexadecimal information, a full decode, or a brief decode. These formats correspond to the three panes described in [“Viewing Captured Packets” on page 226](#).
 - ◆ Choose whether to print all packets, a range of packets, or only the filtered packets.
- 4 Click OK.

Analyzing Traffic Generated by Protocols in Your Network

ZfS lets you determine the distribution of protocols in your network and provides statistical information of the protocols discovered by the RMON2 agent in the network, as well as transport and application layers. You can also add supported and custom protocols to your network. Supported protocols are those that the RMON2 agent is able to decode and count the number of packets transmitted in your network using the protocol. Custom protocols are not supported by the RMON2 agent but are used by nodes in your network.

The following sections explain how you can use ZfS to manage protocols in your network:

- ◆ [“Displaying a List of Protocols Used in Your Network” on page 232](#)
- ◆ [“Determining the Distribution of Protocols in a Segment” on page 234](#)

Displaying a List of Protocols Used in Your Network

You can use the Protocol Directory property page to view a hierarchical representation of supported and custom protocols used in the network, transport, and application layers in your network. By default, the page displays the Protocol Directory Tree that displays a collapsed list of protocols. The protocols used in the data link layer are displayed at the top level. You can expand each protocol to display the list of supported and custom protocols under the selected protocol.

You can also use the Protocol Directory property page to add or delete the protocols supported by the RMON2 agent. For details, see [“Adding Supported Protocols to the Protocol Directory Tree” on page 233](#). The custom protocols that are used by the nodes in your network but are not supported by the RMON2 agent can also be added using the limited extensibility feature of RMON2. For details, see [“Adding Custom Protocols to a Supported Protocol Tree” on page 234](#).

For details about the limited extensibility feature, see [RFC 2021 \(http://www.isi.edu/in-notes/rfc2021.txt\)](http://www.isi.edu/in-notes/rfc2021.txt).

For a selected protocol, you can specify the RMON2 groups you want the RMON2 agent to support. This will let you obtain the RMON2 details of the groups that you specify the agent to support. While adding the protocol, you can enable the agent support for the Host group, Matrix group, and Address Map group. The Groups Supported box in the lower portion of the property page indicates whether the agent support for the Host and Matrix groups in the network layer and application layer, and support for the Address Map group are enabled, disabled, or not supported for the selected protocol. You can configure the values displayed in the Groups Supported box.

The Add and Remove buttons are enabled only when you select a protocol in the Protocol Directory tree.

IMPORTANT: The Traffic Analysis Agent for NetWare and Traffic Analysis Agent for Windows NT/2000 do not support enabling of the Address Map, Host, and Matrix groups for protocols in the Protocol Directory.

To open the Protocol Directory property page:

- 1 Click RMON2 under Service within a node from ConsoleOne.
- 2 Click File > Properties > the Protocol Directory tab.

Refer to the following sections:

- ♦ [“Adding Supported Protocols to the Protocol Directory Tree” on page 233](#)
- ♦ [“Adding Custom Protocols to a Supported Protocol Tree” on page 234](#)

Adding Supported Protocols to the Protocol Directory Tree

Supported protocols are those that the RMON2 agent is able to decode and count the number of packets transmitted in your network using the protocol.

Default values are provided for the parameters of protocols supported by the RMON2 agent. When you enter the name of a protocol, the default values are displayed if the protocol is supported.

To add a protocol to the Protocol Directory tree:

- 1 Open the Protocol Directory property page.
- 2 Select a protocol from the Protocol Directory tree.
- 3 Click Add.

The following table describes the parameters for a selected protocol.

IMPORTANT: The Protocol Name parameter cannot be configured. If you configure the port number or protocol code of a selected protocol, all child protocols of the selected protocol will be deleted.

Parameter	Description
Protocol Name	Displays the name of the protocol.
Protocol ID	Displays the identifier for the protocol. Displays the port number for an application layer protocol or the protocol code for protocols in other layers. The protocol identifier is always a decimal value.
Description	Displays a short description of the selected protocol.
Groups Supported	Displays whether the agent support of the Address Map group, Host group, or Matrix group is enabled for the selected protocol.

If the protocol name you enter or select from the Protocol Name list is supported by the RMON2 agent, the default parameters for the protocol are displayed in the appropriate fields of the Add Protocol dialog box. You cannot edit the parameters once you have added, if you do not want to use the default values.

4 Click OK.

The new protocol is added as a child protocol of the selected protocol. You cannot edit the parameters of the protocol you have added. You would need to delete the protocol and add the protocol again with different parameters.

Adding Custom Protocols to a Supported Protocol Tree

Custom protocols are those that are not supported by the RMON2 agent but are used by nodes in your network. If the RMON2 agent supports the limited extensibility feature of RMON2 for a selected protocol, you can add custom protocols under the selected protocol. See [RFC 2021 \(http://www.isi.edu/in-notes/rfc2021.txt\)](http://www.isi.edu/in-notes/rfc2021.txt) for more information. If the RMON2 agent does not support the limited extensibility feature for a protocol, you cannot add custom protocols under that protocol. A custom protocol cannot have child protocols.

Because default values are not provided for custom protocols, you must enter the appropriate values if you are adding a protocol that is not supported by the RMON2 agent.

To add a custom protocol to the Protocol Directory tree:

- 1** Select a supported protocol from the Protocol Directory tree.
- 2** Click Add.
- 3** In the Protocol Name field, enter the name of the protocol.
- 4** In the Protocol ID field, enter the port number for an application layer protocol or a protocol code for protocols in other layers.
IMPORTANT: The port number or protocol code should be a decimal value.
- 5** From the Groups Supported box, select the groups you want the RMON2 agent to support for the protocol.

The custom protocol is added as a child protocol of the supported protocol.

To remove a protocol from the Protocol Directory tree:

- 1** Select a protocol from the Protocol Directory tree.
- 2** Click Remove.

IMPORTANT: If you remove a protocol that has child protocols, all the child protocols are also removed from the Protocol Directory tree.

Determining the Distribution of Protocols in a Segment

ZFS lets you determine the distribution of protocols discovered by the RMON2 agent. You can use the information displayed in this view to analyze the traffic in your network and to troubleshoot network problems. Use the Protocol Directory property page to add, delete, or edit a protocol. See [“Adding Supported Protocols to the Protocol Directory Tree” on page 233](#) and [“Adding Custom Protocols to a Supported Protocol Tree” on page 234](#) for details.

The distribution of protocols discovered by the RMON2 agent is displayed in the Protocol Distribution view, based on the layer in which the protocols are discovered.

To view the distribution of protocols in the selected segment:

- 1 Select a segment from ConsoleOne.
- 2 Click View > Protocol Distribution.

The view displays the following three tables that list the protocols discovered in the network:

- ◆ Network layer table
- ◆ Transport layer table
- ◆ Application layer table

The protocols discovered by the RMON2 agent are placed in the appropriate table in the Protocol Distribution view depending on the layer in which they were discovered. Each table displays protocol statistics that are updated every 15 seconds.

The following table describes the protocol statistics displayed in the Protocol Distribution view.

Statistic	Description
Protocol Name	The name of the protocol
Packets/s	The average number of packets transmitted per second using the protocol discovered by the agent on the monitored segment
Bytes/s	The average number of bytes transmitted per second using the protocol discovered by the agent discovered on the monitored segment
Packet Rate %	The percentage of packets transmitted using the protocol; this is relative to the total percentage of packets transmitted using all protocols discovered by the agent
Byte Rate %	The percentage of bytes transmitted using the protocol; this is relative to the total percentage of bytes transmitted using all protocols discovered by the agent

IMPORTANT: Only one entry of each protocol is displayed in the Protocol Distribution view. Consolidated statistics are displayed for a supported protocol in more than one protocol suite.

Analyzing Traffic on Switches

ZfS provides statistical information about ports in a monitored switch and a list of nodes connected to each port in your switched network. This information is displayed in the Unified Port Traffic view. You can use the view to determine the load on the desktop and workgroup switches in your switched network. When only one node can be connected to each port in a switch, the switch is known as a desktop switch. When one port of a switch is connected to a connecting device to which more than one node is connected, the switch is called a Workgroup switch.

Ports and nodes connected to ports of a switch can be monitored using an embedded RMON agent or external RMON agent. An embedded RMON agent is installed on the port of a switch. An external RMON agent is installed on a node connected to a switch.

The following sections explain how you can obtain information about switch ports and nodes connected to ports in your switched network:

- ◆ [“Viewing Statistics for Ports in a Switch” on page 236](#)
- ◆ [“Viewing the Summarized Switch Information” on page 236](#)

Viewing Statistics for Ports in a Switch

You can use the Unified Port Traffic view to obtain statistical information about every switch port in your network. The view also displays a drop-down list of nodes connected to each port. The information displayed in this view is useful if you want to troubleshoot a port.

The Unified Port Traffic view displays a list of nodes connected to ports on the switch and statistics for each port. You can view Ethernet specific statistics for Ethernet ports on a switch. Statistics specific to FDDI and token ring ports are not displayed with this version of ZfS, although general port statistics are displayed for all ports on a switch regardless of the media type. You can choose to display all statistics or configure the Unified Port Traffic view to display selected statistics. For details, see [“Choosing Statistics to Display in the Unified Port Traffic View” on page 242](#).

To display the statistics of ports in a switch:

- 1 Select Switch/Bridge under Services within a switch from ConsoleOne.
- 2 Click View > Port Traffic.

Viewing the Summarized Switch Information

The Switch Summary view provides brief information about a selected switch. You can view static information about a selected switch and information about alarms generated on the switch. You can also determine the packets and broadcasts received by the switch per second.

To view the summarized switch information:

- 1 Select Switch/Bridge under Services within a switch from ConsoleOne.
- 2 Click View > Switch Summary.

The Switch Summary view displays static information about a selected switch, as described in the following table.

Statistic	Explanation
Vendor	Name of the switch vendor
Switch Type	Type of switch: Transparent or Source Route
Number of Ports Active	Number of active ports on the switch
Forwarding Table Overflow Count	Number of times the forwarding table has exceeded its capacity
Up Time	Time since the switch was last rebooted
Number of Ports Present	Number of ports present on the selected switch
Number of MAC Addresses Learned	Number of MAC addresses dynamically discovered by the switch

The Switch Summary view displays information about alarms generated on a selected switch, as described in the following table.

Statistic	Explanation
Severity	Severity level attributed to the trap.

Statistic	Explanation
From	Network address of the device that sent the alarm to the alarm management system.
Owner	Segment or device affected by the alarm.
Summary	Summary of the event, often including the name or address of the object affected by the alarm.
Received Time	Date and time when the alarm management system received the alarm.
Type	Generic description of the alarm. For example, Volume out of disk space.
Category	Displays the category of the alarm based on the MIB that defines the trap-type objects. The category is directly related to the MIBs included in the management server MIB pool. For example, the category for NetWare servers is based on the NetWare Server Alarm MIB.

The Switch Summary view displays dynamic information about a selected switch, as described in the following table.

Statistics	Explanation
Switch Load (pkts/sec)	The load on the switch based on packets received by the switch per second
Frames Dropped/sec	The number of received packets discarded per minute
Broadcasts/sec	The number of broadcasts received by the switch from the nodes connected to ports of the switch

Optimizing Traffic Analysis

The tools provided by ZfS to analyze your network performance have default settings. You can change the default settings of various views to display only the information you require.

The following sections provide information about how you can configure the ZfS tools to suit your networking environment:

- ◆ [“Choosing Options to Display Stations on a Segment” on page 237](#)
- ◆ [“Choosing Options to Display Trend Statistics” on page 239](#)
- ◆ [“Choosing Options to Display the Top Nodes Graph” on page 241](#)
- ◆ [“Choosing Statistics to Display in the Unified Port Traffic View” on page 242](#)
- ◆ [“Choosing Options to Display a Captured Packet” on page 243](#)
- ◆ [“Configuring Alarm Options from the Set Alarm Dialog Box” on page 244](#)
- ◆ [“Configuring the Monitor Nodes for Inactivity View” on page 246](#)

Choosing Options to Display Stations on a Segment

You can configure the Stations view to display only the top 20 nodes or all nodes on the monitored segment. You can also choose the statistic based on which you want to display the top 20 nodes.

The following configuring options are available:

- ◆ “Displaying Statistics for All Nodes on a Segment” on page 238
- ◆ “Displaying Statistics for the Top 20 Nodes on a Segment” on page 238
- ◆ “Choosing a Statistic Based on Which Top 20 Nodes Are Displayed” on page 238

Displaying Statistics for All Nodes on a Segment

To display statistics for all nodes on a segment:

- 1 Select a segment from ConsoleOne.
- 2 Click View > Stations.

To display all nodes on a segment, more time is required and more network traffic is generated.

- 3 From the Stations view, click View > Show All Stations.

Displaying Statistics for the Top 20 Nodes on a Segment

To display statistics for the top 20 nodes on a segment:

- 1 Select a segment from ConsoleOne.
- 2 Click View > Stations.
- 3 From the Stations view, click View > Show Top N Stations.

Choosing a Statistic Based on Which Top 20 Nodes Are Displayed

Packets out per second is the default statistic based on which top 20 nodes are displayed in the Stations view. To choose a different statistic based on which you want the top 20 nodes to be displayed, do either of the following:

- ◆ From the Stations view, click View > Show Top N Stations > choose a statistic from the list of statistics displayed.
- ◆ Click the Top Nodes Statistics drop-down box in the toolbar of the Stations view > choose a statistic from those displayed.

The available statistics are described in the following table.

Statistic	Explanation
Packets/s In	Packets per second received by a node
Packets/s Out	Packets per second transmitted by a node
Bytes/s In	Bytes per second received by a node
Bytes/s Out	Bytes per second transmitted by a node
Errors/s	Errors per second transmitted by a node
Broadcasts/s	Broadcast packets per second transmitted by a node
Multicasts/s	Multicast packets per second transmitted by a node (packets transmitted to a specific group of nodes)

If you close the Stations view after changing the default settings, you will be prompted to save the changes made to the default settings. If you want the Stations view to be displayed based on the statistic you chose, you can save the setting. The next time you open ConsoleOne and launch the Stations view, you will be able to view the nodes on the monitored segment based on the statistic you specified.

Choosing Options to Display Trend Statistics

You can change the default settings based on which the segment performance trends are displayed in the Segment Trends view.

The following configuration options are available:

- ◆ [“Choosing Statistics Based on Which Trend is Displayed” on page 239](#)
- ◆ [“Setting the Time-Scale Options” on page 241](#)

Choosing Statistics Based on Which Trend is Displayed

To change the statistics based on which segment performance trend is displayed:

- 1 Click the Profile button in the Segment Trends view.
- 2 Select a profile from the Select Profile list.

The default profile will display a trend with statistical information of total packets, good packets, and error packets on the monitored segment.

If you choose not to use the profiles listed in the Select Profile list, you can select the required statistics from the Select Statistics list.

The statistics list lets you examine the Ethernet, FDDI, and token ring statistics described in the following table.

Statistic	Media Support	Explanation
Abort Delimiter Errors/s	Token ring	Average number of abort delimiter errors observed per second. This error indicates that a node aborts a transmission.
AC Errors/s	Token ring	Average number of AC errors observed per second. This error is reported when an intended recipient of a packet fails to mark it as received or flags an error on it.
Beacons	FDDI and token ring	Average number of beacons per second observed in the sampling interval. A station transmits these packets when it detects a hard failure upstream.
Broadcast Packets/s	Ethernet, FDDI, token ring	Number of broadcast packets per second.
Burst Errors/s	Token ring	Average number of burst errors observed per second. This error indicates that a node detects the absence of transitions for the required time.
Claim Tokens/s	FDDI ring	Average number of times that the ring enters the claim token state from the normal ring state or ring purge state per second.

Statistic	Media Support	Explanation
CRC/Alignment Errors/s	Ethernet and FDDI ring	Number of cyclic redundancy check (CRC)/alignment errors per second.
Echo Pkts/s	FDDI ring	Average number of echo frames received on the network per second.
Elasticity Buffer Errors/s	FDDI ring	Average number of elasticity buffer overflow errors reported by this station per second. This is due to the difference in the clock frequency between the transmitting and receiving stations.
Error Packets/s	Ethernet	Number of error packets per second.
Fragments/s	Ethernet	Number of fragments per second.
Frame Copied Errors/s	FDDI ring	Average number of frame copied error frames reported per second by the station.
Frequency Errors/s	Token ring	Average number of frequency errors observed per second. This error indicates that a token ring clock on a node differs too much from the clock on the active monitor.
Good Packets/s	Ethernet	Number of good packets per second.
Internal Errors/s	Token ring	Average number of internal errors observed per second. These errors generally indicate a network board failure.
Jabbers/s	Ethernet	Number of jabbers per second.
Line Errors/s	Token ring	Average number of line errors observed per second. These packets are of valid size but have a faulty Frame Check Sequence (FCS) and do not end on an 8-bit boundary.
Lost Frames/s	FDDI and token ring	Average number of lost frame errors on the network observed per second.
Monitor Contentions/s	Token ring	Average number of monitor contentions observed per second; these packets are transmitted by all active nodes when no active monitor is detected on the ring.
Multicast Packets/s	Ethernet, FDDI, and token ring	Number of multicast packets per second.
Oversize Packets/s	Ethernet	Number of oversize packets per second.
Packets	FDDI and token ring	Average number of packets observed per second in the sampling interval.
Receive Congestion Errors/s	Token ring	Average number of receive congestion errors observed per second. This error indicates that a node recognizes a frame addressed to its address, but has no available buffer space.

Statistic	Media Support	Explanation
Ring Wraps/s	FDDI ring	Average number of times a wraparound condition has been detected at this interface per second. This entry does not indicate the number of times the ring has actually wrapped around. It only indicates the number of times the ring has wrapped around this physical path.
Token Errors/s	Token ring	Average number of token errors observed per second. This error indicates that a token is corrupted or the active monitor did not see a new frame in the required amount of time.
Total Bytes/s	Ethernet	Average number of total bytes per second.
Total Packets/s	Ethernet	Average number of total packets per second.
Undersize Packets/s	Ethernet	Number of undersize packets per second.
Unicast Packets/s	Ethernet	Number of unicast packets per second.
Utilization%	Ethernet, FDDI, and token ring	Percentage of maximum network capacity used by all packets in the sampling interval.

If you close the Segment Trends view after changing the default statistics based on which trend is displayed, you will be prompted to save the changes made to the default settings. If you want the segment performance trend to be displayed based on the profile or statistics you chose, you can save the settings that you define. The next time you open ConsoleOne and launch the Segment Trends view, you will be able to view the trend based on the profile or statistics you defined.

Setting the Time-Scale Options

The segment performance trend is updated once every minute. You can set a different time scale based on which you want to update a graph. Select from the following time-scale options:

- ◆ Real Time
- ◆ One Hour
- ◆ One Day
- ◆ One Week
- ◆ One Month
- ◆ One Year

TIP: If you close the Segment Trends view after changing the default time-scale option based on which trend is displayed, you will be prompted to save the changes made to the default settings. If you do not want the trend to be updated in real time, you can save the time-scale setting you choose. The next time you open ConsoleOne and launch the Segment Trends view, the trend will be updated based on the time-scale option you selected.

Choosing Options to Display the Top Nodes Graph

You can configure the Segment Dashboard view to display or disable the top nodes graph. For details, see [“Viewing the Graph of the Top Nodes on a Monitored Segment” on page 211](#). The top nodes graph is displayed in the lower portion of the Segment Dashboard view. Packets out per

second is the default statistic based on which the graph is displayed. You can choose a different statistic based on which you want the graph to be displayed.

The following configuring options are available:

- ◆ [“Displaying the Top Nodes Graph in the Segment Dashboard View” on page 242](#)
- ◆ [“Choosing the Statistic Based on Which Top Nodes Graph Is Displayed” on page 242](#)
- ◆ [“Disabling the Top Nodes Graph in the Segment Dashboard View” on page 242](#)

Displaying the Top Nodes Graph in the Segment Dashboard View

To display the top nodes graph in the Segment Dashboard view:

- 1 From the Segment Dashboard view, click View > Show Top N Graph.

Choosing the Statistic Based on Which Top Nodes Graph Is Displayed

To display the top nodes graph based on a different statistic, do either of the following from the Segment Dashboard view:

- ◆ Click View > Show Top N Graph > choose a statistic.
- ◆ Click the Top Nodes Statistics drop-down box in the toolbar of the Segment Dashboard view > select a statistic.

The statistics are described in the following table.

Statistic	Explanation
Broadcasts/min	Broadcast packets per minute transmitted by a node
Bytes/s in	Bytes per second received by a node
Bytes/s out	Bytes per second transmitted by a node
Errors/min	Errors per minute transmitted by a node
Packets/s in	Packets per second received by a node
Packets/s out	Packets per second transmitted by a node
Multicasts/min	Multicast packets per minute transmitted by a node

IMPORTANT: Errors per minute, broadcasts per minute, and multicasts per minute are updated every 60 seconds rather than every 5 seconds.

Disabling the Top Nodes Graph in the Segment Dashboard View

To disable the top nodes graph in the Segment Dashboard view:

- 1 From the Segment Dashboard view, click View > Disable Top N Graph.

Choosing Statistics to Display in the Unified Port Traffic View

ZFS provides statistics for each port on the switch. You can view port statistics and a list of nodes connected to each port using the Unified Port Traffic view. You can view Ethernet-specific statistics for Ethernet ports on a switch. Although statistics specific to FDDI and token ring ports

will not be displayed with this version of ZfS, general port statistics are displayed for all ports on a switch regardless of the media type. For details, see [“Viewing Statistics for Ports in a Switch” on page 236](#). You can choose to display only the selected statistics in the Unified Port Traffic view.

To select statistics to be displayed in the Unified Port Traffic view:

- 1 From the Unified Port Traffic view, click View > Settings.
- 2 Click the statistics from the Available Columns list > click Add.

The following table describes the general port statistics displayed for a port, regardless of the media type of the port.

Statistic	Explanation
Frames In/sec	Number of frames received by the port per second.
Frames Out/sec	Number of frames sent by port per second.
Port Link Status	Displays if the port is active or inactive. If the port is active, it can transmit and receive packets.
Speed	The speed at which packets are transmitted or received by the port.
Media Type	Media type of the selected port.
Local Traffic	Rate of traffic going towards nodes on the same port.

The following table describes the Ethernet-specific statistics displayed for an Ethernet port in addition to the general port statistics listed above.

Statistic	Explanation
Collisions/sec	Number of collisions per second
Utilization	Percentage of maximum network capacity currently consumed by packet traffic on the port
Broadcasts/sec	Number of broadcast packets per second currently received and sent by the port
Multicasts/sec	Multicast packets per second received and sent by the port
Packets/sec	Number of packets per second received and sent by the port
CRC Align Error	Total number of line errors reported by the port
Oversize Pkts	Number of oversize packets received and sent by the port

Choosing Options to Display a Captured Packet

ZfS provides default settings to display a captured packet in the Trace Display window.

To change the default settings and display the trace differently:

- 1 Open the Trace Display window.
- 2 From the Trace Display menu, click View > Options.
- 3 Select how you want to display the decoded packet.

- ◆ Full Protocol Decode: Provides information about each field in each protocol layer in a selected packet. This is the default decoding.
 - ◆ One Line Per Protocol Layer: Provides a line of information for each protocol layer of a selected packet.
- 4** Select the level at which you want to display the initial highlight position.
 - ◆ At Highest Protocol Layer: Places the initial highlighting at the highest protocol layer in a packet. This is the default.
 - ◆ At Packet Header: Places the initial highlighting at the packet header.
 - 5** Select the format in which you want to display the decoded packet.
 - ◆ ASCII: Displays the hex data in ASCII format. This is the default.
 - ◆ EBCDIC: Displays the hex data in EBCDIC format.

Configuring Alarm Options from the Set Alarm Dialog Box

ZfS provides default alarm threshold values for a segment. You can set threshold values for various error conditions on Ethernet, FDDI, and token ring segments to eliminate the need to constantly monitor the segments.

When a segment alarm is enabled, the RMON agent monitors the segment based on the alarm threshold settings. If the configured threshold value is exceeded, the RMON agent sends a trap to the management server, which forwards it to ConsoleOne.

You should change the default values for alarm thresholds as appropriate for your organization. You can determine the appropriate value by observing average and peak traffic levels on your network using the Segment Trends view. For details, see [“Analyzing Trend Data for a Segment” on page 211](#). You can do this as a part of creating a baseline of typical segment activity on your network.

To set an alarm threshold for a segment:

- 1** Select a segment from ConsoleOne.
- 2** Click File > Properties > the Segment Alarms tab.
- 3** Select a segment statistic > click Edit.
- 4** Click Enable to enable the alarms set for the monitored segment.

When you click Enable, the text fields and the Default button will be enabled. However, if the default threshold values are not found, the Default button will not be enabled.

- 5** Enter the threshold value.
- 6** Specify the sampling time interval.

The RMON agent uses the sampling time interval to average the statistic to determine whether the alarm threshold was exceeded.

TIP: You can also use the Segment Dashboard view to define alarm threshold values for segment statistics. For details, see [“Defining Alarm Thresholds for Statistics Displayed in the Segment Dashboard View” on page 211](#).

The following table describes the alarm statistics that ZfS tracks for Ethernet, FDDI, and token ring segments.

Statistic	Media Support	Explanation
Abort Errors	Token ring	Average number of abort errors observed per second in the sampling interval. These errors resemble line errors, but occur in the middle of a transmission.
AC Errors	Token ring	Average number of Address Recognition (and Frame Copied) errors observed per second in the sampling interval. This error is reported when an intended recipient of a packet fails to mark it as received or flags an error on it.
Beacons	FDDI and token ring	Average number of beacons per second observed in the sampling interval. A station transmits these packets when it detects a hard failure upstream.
Broadcasts	Ethernet, FDDI, and token ring	Average number of packets per second sent to the broadcast address FF-FF-FF-FF-FF-FF. Broadcast messages typically consist of general requests for information or transmission of status information to all stations.
Burst Errors	Token ring	Average number of burst errors observed per second in the sampling interval. A burst error is caused by a lack of signal transitions between stations for a short period of time.
Claim Tokens	FDDI ring	Average number of times that the ring enters the claim token state from the normal ring state or ring purge state per second.
Congestion Errors	Token ring	Average number of congestion errors observed per second in the sampling interval. The receiving station runs out of buffer space to store the packet.
CRC Errors	Ethernet and FDDI ring	Average number of CRC errors observed per second in the sampling interval. These packets are of valid size but have a faulty FCS.
Echo Pkts	FDDI ring	Average number of echo frames received on the network per second.
Elasticity Buffer Errors/s	FDDI ring	Average number of elasticity buffer overflow errors reported per second by this station. This is due to the difference in the clock frequency of the transmitting and receiving stations.
Fragments	Ethernet	Average number of fragments observed per second in the sampling interval. Fragments are packets that contain fewer than 64 bytes and have a faulty FCS. They are typically a result of collisions.
Frame Copied Errors	FDDI and Token ring	Average number of frame copied errors observed per second in the sampling interval. This error indicates that a station has detected that another station accepted a packet addressed to the first station.
Frequency Errors	Token ring	Average number of frequency errors observed per second in the sampling interval. This error indicates that a token ring clock on a station differs from the clock on the active monitor.
Internal Errors	Token ring	Average number of internal errors observed per second in the sampling interval. These errors generally indicate a network adapter board failure.

Statistic	Media Support	Explanation
Jabbers	Ethernet	Average number of jabber packets observed per second in the sampling interval. A jabber consists of packets that contain more than 1518 bytes and have a faulty FCS.
Line Errors	Token ring	Average number of line errors observed per second in the sampling interval. These packets are of legal size but have a faulty FCS and do not end on an 8-bit boundary.
Lost Frames	FDDI and token ring	Total number of lost frame errors received on the network. A lost frame error indicates that the end delimiter of a frame is lost in the network.
Monitor Contentions	Token ring	Average number of monitor contentions observed per second in the sampling interval. These packets are transmitted when no active monitor is detected on the ring.
Multicasts	Ethernet, FDDI, and token ring	Average number of packets per second sent to multicast addresses.
Oversize	Ethernet	Average number of oversized packets observed per second in the sampling interval. Oversized packets contain more than 1518 bytes, including the FCS.
Packets	Ethernet, FDDI, and token ring	Total number of packets observed per second in the sampling interval.
Ring Wraps/s	FDDI ring	Average number of times a wraparound condition has been detected at this interface per second. This entry does not indicate the number of times that the ring has actually wrapped around. It only indicates the number of times the ring has wrapped around this physical path.
Token Errors	Token ring	Average number of token errors observed per second in the sampling interval. This error indicates that a token is corrupted or the active monitor did not detect a new frame transmitted during the current sampling interval.
Undersize	Ethernet	Average number of undersized packets observed per second in the sampling interval. Undersized errors are shorter than 64 bytes.
Utilization(%)	Ethernet, FDDI, and token ring	Percentage of maximum network capacity used by all packets in the sampling interval.

When you have set the appropriate threshold values for the segments in your network, you can use the Save As Default button on the Segment Alarms property page to save the values you defined as the default values. However, the default threshold values provided by ZfS will not be available once you apply the new values.

Configuring the Monitor Nodes for Inactivity View

By default, the poll interval for refreshing the Monitor Nodes for Inactivity view is zero seconds. You can configure the poll interval based on which you want the view to be refreshed. The agent monitoring nodes on a monitored segment declares a node as inactive after verifying it for a specified period of time. You can change the time duration for the agent to verify the node before declaring it inactive.

The following configuring options are available:

- ◆ “Specifying the Poll Interval for Refreshing the Monitor Nodes for Inactivity View” on page 247
- ◆ “Specifying the Duration for the Agent to Determine if a Node Is Inactive” on page 247

Specifying the Poll Interval for Refreshing the Monitor Nodes for Inactivity View

You can modify the PollInterval parameter in the LSMPARAMETERS.PROPERTIES file to specify the poll interval for refreshing the Monitor Nodes for Inactivity view.

To specify a poll interval for refreshing the Monitor Nodes for Inactivity view:

- 1** Open the LSMPARAMETERS.PROPERTIES file located in the *operating_system_drive*\INSTALL\CONSOLEONE\BIN directory.

- 2** Specify a value for the PollInterval parameter.

The PollInterval value should be a positive value, in seconds. The default value is zero (0) seconds.

Specifying the Duration for the Agent to Determine if a Node Is Inactive

When a selected node becomes inactive, the agent monitoring the node verifies the state of the node for one minute before declaring it inactive. You can modify the HostTimeOut parameter in the LSMPARAMETERS.PROPERTIES file to change the duration for the agent to verify the selected node before declaring it inactive. The agent verifies the inactive node for the specified period of time before declaring it inactive.

To change the duration for the agent to verify a node before declaring it inactive:

- 1** Open the LSMPARAMETERS.PROPERTIES file located in the *operating_system_drive*\INSTALL\CONSOLEONE\BIN directory.

- 2** Specify a value for the HostTimeOut parameter.

The HostTimeOut value should be a positive value, in minutes. The default value is one (1) minute.

Understanding the Traffic Analysis Agents

Traffic Analysis agents enable you to monitor a heterogeneous LAN environment comprised of Ethernet, FDDI, and token ring segments from the easy-to-use ZfS interface.

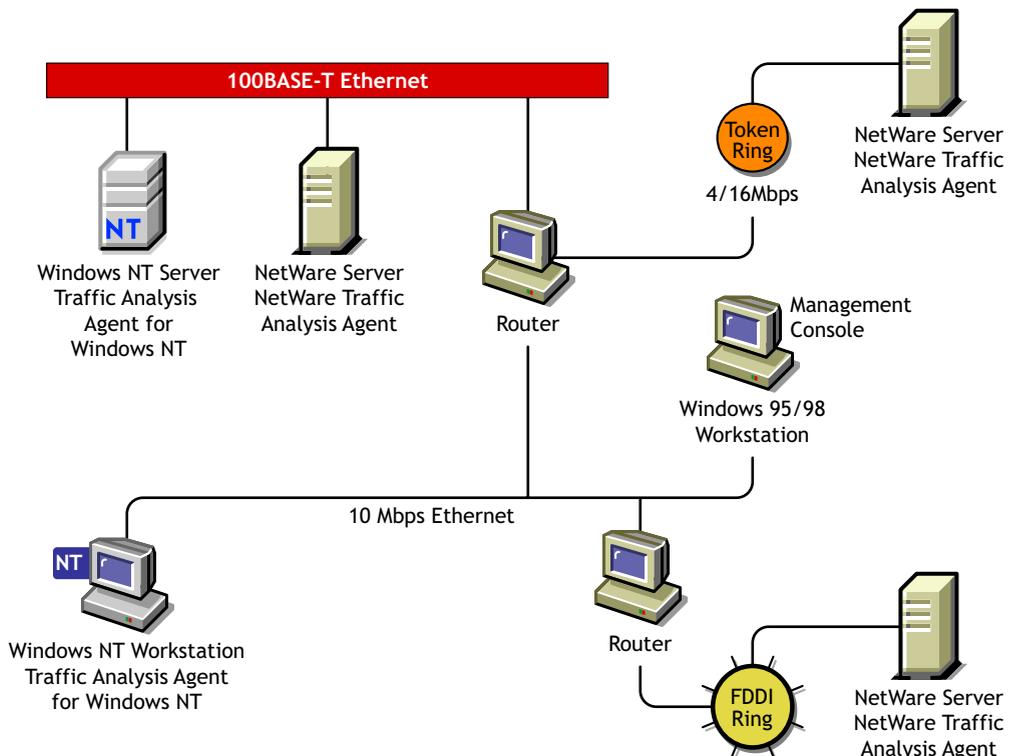
Traffic Analysis agents are RMON agents that can run on a NetWare server, Windows NT/2000 server, or a Windows NT workstation. They implement a set of functionality defined by the RMON MIB (RFC 1757 (<http://www.isi.edu/in-notes/rfc1757.txt>)). These agents collect information about activity on your network and make it available to ConsoleOne via SNMP.

The following functionality is provided by the Traffic Analysis Agents:

- ◆ Monitor the performance of segments and provide vital network statistical information to ConsoleOne
- ◆ Make it easy to set alarm thresholds for proactive network management
- ◆ Capture all packets or selected packets to help you diagnose and resolve problems on the monitored networks

- ◆ Monitor multiple network segments including the Symmetric Multi-Processing (SMP) architecture
- ◆ Monitor network segments for problems, such as high network utilization and communication errors
- ◆ Track dynamic IP address assignments from the DHCP server to the nodes on the network
- ◆ Store data to display real-time trends (hourly) and historical trends (daily, weekly, monthly, and yearly) for statistics such as Total Bytes, Total Packets, Good Packets, Error Packets, and so forth
- ◆ Monitor nodes for inactivity, so that you are alerted if the monitored nodes becomes inactive

The following figure illustrates the functionality of traffic analysis agents.



ZfS includes the following traffic analysis agents:

- ◆ Traffic Analysis Agent for NetWare.
For details, see [“Using the Traffic Analysis Agent for NetWare” on page 249.](#)
- ◆ Traffic Analysis Agent for Windows NT/2000.
For details, see [“Using the Traffic Analysis Agent for Windows NT/2000” on page 263.](#)

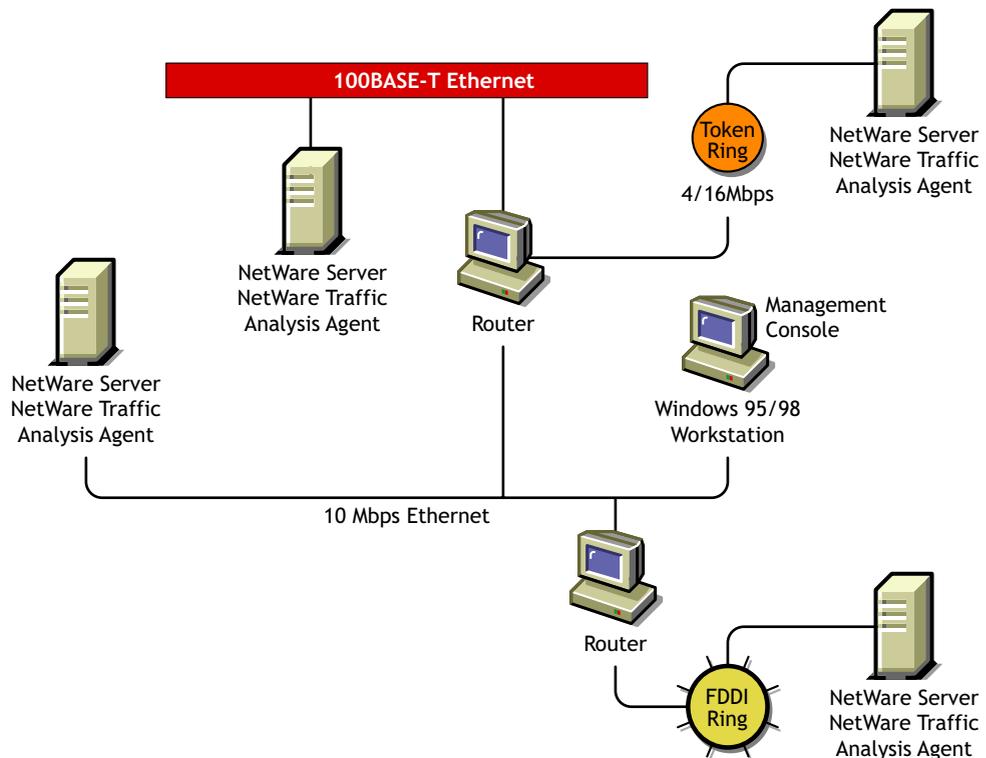
The ZfS traffic analysis agents are RMON Plus agents. For details, see [“Functionality of RMON Plus Agents” on page 197.](#) These agents also implement the first two RMON2 groups. The first RMON2 group is the Protocol Directory group, which provides a table of protocols for which the agent will monitor and maintain statistics. The second RMON2 group is the Protocol Distribution group, which provides a table of statistics for each protocol in the directory. For details, see [“Functionality of RMON2 Agents” on page 198.](#)

Using the Traffic Analysis Agent for NetWare

The Traffic Analysis Agent for NetWare (NLA 1.30) runs on a NetWare server. It is a set of NLM programs that enable NetWare 4.x and 5.x to monitor traffic on Ethernet, FDDI, or token ring segments.

The Traffic Analysis Agent for NetWare implements token ring extensions for the RMON MIB (RFC 1513 (<http://www.isi.edu/in-notes/rfc1513.txt>)) for token ring media, and a Novell proprietary MIB for FDDI media, in addition to implementing an RMON (RFC 1757 (<http://www.isi.edu/in-notes/rfc1757.txt>)) for Ethernet media. The Traffic Analysis Agent for NetWare also implements the first two groups for RMON2 (RFC 2021 (<http://www.isi.edu/in-notes/rfc2021.txt>)).

The following figure illustrates a functional view of the Traffic Analysis Agent for NetWare:



The following sections provide information about optimizing and using the Traffic Analysis Agent for NetWare:

- ♦ “Planning to Install the Traffic Analysis Agent for NetWare” on page 249
- ♦ “Optimizing the Traffic Analysis Agent for NetWare Performance” on page 250
- ♦ “Using the Console Utility of the Traffic Analysis Agent for NetWare” on page 257

Planning to Install the Traffic Analysis Agent for NetWare

To successfully install the Traffic Analysis Agent for NetWare on a NetWare server, the server must meet the system requirements specified in “[Installing and Setting Up Management and Monitoring Services](#)” in the *Installation* guide.

You should configure NetWare SNMP parameters as explained in [Chapter 13, “Using SNMP Community Strings,” on page 303](#). This will ensure a smooth installation of the Traffic Analysis Agent for NetWare on the server.

NOTE: Although it is not required, it is recommended that you uninstall previous versions of the LANalyzer Agent (referred to as the Traffic Analysis Agent in ZfS). If you do not uninstall the previous version of the agent, you must verify that the upgraded NetWare servers run the new Traffic Analysis Agent.

Optimizing the Traffic Analysis Agent for NetWare Performance

The measures described in the following sections can improve the performance of your Traffic Analysis Agent for NetWare servers.

You can configure the Traffic Analysis Agent for NetWare functions described in the following sections by setting the parameters in the LANZ.NCF file.

- ◆ [“Contents of the LANZ.NCF File” on page 250](#)
- ◆ [“Modifying the LANZ.NCF File” on page 254](#)

Contents of the LANZ.NCF File

The LANZ.NCF file loads all the NLM software required for the Traffic Analysis Agent for NetWare operation. The LANZ.NCF file resides in the SYS:\Zfs_agnt\lanz directory.

The following example displays the complete text of the default LANZ.NCF file.

```
#
# NetWare LANalyzer Agent
# Version 1.3
#
# - - - - -
# LANZ.NCF: NetWare LANalyzer Agent Load File
#
# This NCF file is created by the NetWare LANalyzer Agent install program.
# It is used to load the NetWare Loadable Module files that make up NetWare
# LANalyzer Agent.
# WARNING: You should not modify this file unless you need to change one of
# the configuration parameters documented below. Other changes to this
# file are not recommended. Should you damage this file, you must reinstall
# NetWare LANalyzer Agent.
#
# NOTE: To enable or disable the monitoring of network adapters by
# NetWare LANalyzer Agent, use the LANZCON utility as described in the
# NetWare LANalyzer Agent Installation and Administration guide.
#
```

```
# - - - - -
# Load Parameter Descriptions
#
# load LANZSU debug=1
#
# debug=1    Turns on the LANZ Control screen to see the transactional
# messages from the NetWare LANalyzer Agent.
#
# load LANZMEM bound=KB age=HHH
#
# bound=KB   This is the upper limit on memory that can be allocated
# dynamically by the NetWare LANalyzer Agent.
#
# Increasing this number allows you to create larger packet
# capture buffers and maintain data for inactive stations
# for a longer period of time.
#
# Decreasing this value reduces the amount of memory that
# can be used by NetWare LANalyzer Agent. This leaves more
# memory for the other server tasks.
#
# NetWare LANalyzer Agent automatically purges data for
# inactive stations as the memory boundary is approached.
# This allows NetWare LANalyzer Agent to adjust to
#
# the memory that is available to it dynamically.
#
# If the boundary is low, purging occurs frequently, saving
# only data for stations that have been recently active on
# the network. If this happens, a message appears on the
# system console indicating that not enough memory has been
# allocated to NetWare LANalyzer Agent.
#
# KB is the memory boundary in kilobytes.
#
```

```

# Initial value: Set by the installation program
# based on memory usage
#
# Minimum recommended value:      512
#
# Maximum recommended value:      75% of free server memory
# when NLM files are loaded
#
# Default value:                   If bound=KB is not specified,
# it defaults to 3072.
#
# age=HHH      NetWare LANalyzer Agent purges data for stations that have
# not been active on the network recently. This parameter
# controls how long data for inactive stations is maintained.
#
# Memory that is used by the station table is not available
# for other uses, such as capturing packets. Reducing the
# AGE value tends to increase the amount of memory
# available for capturing packets.
#
# If you cannot allocate capture buffers that are large,
# you may need to reduce the AGE value.
#
# HHH is the inactivity period, in hours, before station data
# is purged.
#
# Minimum recommended value:      1
#
# Default value:                   If age=HHH is not specified,
# it defaults to 168 (1 week)
#
# load LANZDI level=1
#
# level=1      It indicates that the LANZDI will stop receiving packets
# when CPU utilization gets high.

```

```

#
# Default is OFF. LANZDI will continue to receive packets even
# when CPU utilization gets high.
#
# load LANZSM topn=N
#
# topn=N      The number of concurrent sorts of top N nodes that
#
# NetWare LANalyzer Agent supports for each network adapter.
#
# Recommended value: 4
# Minimum value:    2
# Maximum value:    10
#
# load LANZTR poll = 1
#
# poll=1      Polls token ring source-routed bridges.
#
# load LANZCTL trapreg=1
#
# trapreg=1 Causes SNMP traps to be sent to management consoles
# advertising themselves on the network, as well as stations
# listed in SYS:\ETC\TRAPTARG.CFG. Omitting this parameter
# or setting it to 0 causes traps to be sent only to those
# stations listed in the SYS:\ETC\TRAPTARG.CFG file.
#
# - - - - -
load gtrend.nlm
load lanzsu.nlm
load lanzmem.nlm bound = 3072 AGE = 168
load lanzlib.nlm
load lanzdi.nlm
load lanzael.nlm
load lanzhis.nlm
load lanzfcb.nlm

```

```
load lanzsm.nlm topn = 4
load lanztr.nlm
load lanzfddi.nlm
load lanzctl.nlm trapreg = 1
```

Modifying the LANZ.NCF File

The following sections describe how to modify the parameters of the commands in the LANZ.NCF file to configure the Traffic Analysis Agent for NetWare functions:

- ◆ [“Turning On the LANZ Control Screen” on page 254](#)
- ◆ [“Disabling Packet Capture” on page 254](#)
- ◆ [“Disabling Generation of Duplicate IP Address Alarms” on page 255](#)
- ◆ [“Setting Packet Flow Control” on page 255](#)
- ◆ [“Setting the Upper Limit of Available Memory” on page 255](#)
- ◆ [“Purging Data from Server Memory” on page 256](#)
- ◆ [“Sorting Concurrent Top Stations” on page 256](#)
- ◆ [“Automatically Sending Alarms to the ZfS Site Server” on page 256](#)
- ◆ [“Polling Source Route Bridges” on page 257](#)
- ◆ [“Activating Changes in the LANZ.NCF File” on page 257](#)

To make changes in the LANZ.NCF file and modify the configuration of the Traffic Analysis Agent for NetWare:

- 1** Open the LANZ.NCF file with a text editor.
- 2** Insert or modify the appropriate parameter as shown and save the file.
- 3** Unload and reload the Traffic Analysis Agent for NetWare, as described in [“Activating Changes in the LANZ.NCF File” on page 257](#).

Turning On the LANZ Control Screen

The LANZ control screen reports significant events for the Traffic Analysis Agent for NetWare.

To turn on the LANZ control screen, insert the DEBUG parameter in the LOAD LANZSU.NLM statement, as shown below:

```
LOAD LANZSU.NLM DEBUG=1
```

The default is Off.

Disabling Packet Capture

You might want to disable packet capture to prevent others from observing sensitive data captured in the packets sent on the network segment.

To disable the packet capture, insert a comment mark (#) in the LOAD LANZFCB statement, as shown below:

```
LOAD LANZFCB.NLM
```

You can also control packet capture during high levels of traffic instead of disabling packet capture entirely. For details, see [“Setting Packet Flow Control” on page 255](#).

Disabling Generation of Duplicate IP Address Alarms

In the DHCP environment, the IP address is released to the DHCP server when a DHCP client is shut down. During the process of releasing the IP address to the DHCP server, the client sends a DHCPRELEASE packet. If this packet does not reach the agent, false duplicate IP address alarms will be generated.

To disable the generation of duplicate IP address alarms, specify zero (0) as the value for the DUPIP parameter, as shown below:

```
LOAD LANZSM DUPIP=0
```

If the DUPIP parameter contains a non-zero value or if the parameter is not specified, duplicate IP address alarms are generated.

Setting Packet Flow Control

The Traffic Analysis Agent for NetWare typically operates in promiscuous mode, receiving all packets on the network. However, if server utilization is high and performance becomes degraded, you can set the LEVEL parameter to 1, which configures the agent to pause when server traffic is high, and then automatically resume operation in promiscuous mode when the traffic level returns to normal.

The default is not to specify the LEVEL parameter at all, which allows continuous operation in promiscuous mode.

To set packet flow control, use the LEVEL parameter setting, as shown below:

```
LOAD LANZDI LEVEL=1
```

Setting the Upper Limit of Available Memory

The BOUND parameter sets the upper limit of available memory that can be allocated dynamically to the Traffic Analysis Agent for NetWare.

The value of the BOUND parameter is measured in kilobytes (KB). The default value is 3072 KB. The minimum recommended value is 512 KB. The maximum recommended value is 75% of the memory that is available after all NLM files are loaded.

You might receive the message "Insufficient memory available for the Traffic Analysis Agent for NetWare" in the following situations:

- ◆ The server has too little memory
- ◆ The server has sufficient memory, but the memory is not available to the Traffic Analysis Agent for NetWare
- ◆ You requested a packet capture buffer that is too large, and the agent granted you less memory than requested

In each case, you should increase the value of the BOUND parameter and add more RAM to your NetWare server.

To change the upper limit of available memory, edit the BOUND parameter, with the appropriate value, as shown below:

```
LOAD LANZMEM BOUND=3072 AGE=168
```

Purging Data from Server Memory

The Traffic Analysis Agent for NetWare holds its data in server memory. You can control the amount of data held in memory by setting the value of the AGE parameter. When data reaches the age specified in the parameter, the data is purged from memory. The AGE parameter is particularly useful on large, bridged networks.

The value of the AGE parameter is measured in hours. The default value is 168, or one week. The minimum recommended value is one hour.

You should lower the AGE parameter if you receive the message "Insufficient memory available for the Traffic Analysis Agent for NetWare" and you have allocated sufficient memory for the agent.

Having insufficient memory is not harmful to the agent or the server. The Traffic Analysis Agent for NetWare can run indefinitely, even when the memory allocated to it is not sufficient.

To modify the amount of data held in server memory, change the value of the AGE parameter, as shown below:

```
LOAD LANZMEM BOUND=3072 AGE=168
```

Sorting Concurrent Top Stations

The Traffic Analysis Agent for NetWare sorts stations whenever the top eight graphs on the Segment Dashboard view, the Stations view, or both are displayed by ConsoleOne. The sorts are independent of each other and can be computed on the basis of different statistics.

Because each of the sort computations uses server CPU cycles, you should limit the number of concurrent computations.

To set the number of concurrent sort computations per network adapter, set the TOPN parameter, as shown below:

```
LOAD LANZSM TOPN=n
```

The default value is 4. The minimum value is 2. The maximum value is 10.

Automatically Sending Alarms to the ZfS Site Server

The Traffic Analysis Agent for NetWare can automatically send SNMP alarms (sometimes referred to as SNMP traps) to the ZfS site server or other nodes on the network in the following configurations:

- ◆ The Traffic Analysis Agent for NetWare receives the SAP packets sent by the ZfS site server
- ◆ The ZfS site server or other node is listed in the server's TRAPTARG.CFG file. This file can be edited to add other trap targets.

The TRAPTARG.CFG file is stored in the SYS:\ETC directory. The file provides instructions for its use. You can edit the file with any ASCII text editor.

To enable alarms to be sent automatically, add the TRAPREG parameter setting, as shown below:

```
LOAD LANZCTL TRAPREG=1
```

The default is 1. If you omit the TRAPREG parameter or set its value to zero (0), the agent sends alarms only to management consoles listed in the TRAPTARG.CFG file.

Polling Source Route Bridges

To control source route bridge polling on token ring networks, use the POLL parameter, as shown below:

```
LOAD LANZTR POLL=1
```

1 = On and 0 = Off.

Setting the POLL parameter to 1 polls source routed bridges once every second. You cannot change the polling rate. The default is On.

To turn off this function, set the POLL parameter to zero (0), as shown below:

```
LOAD LANZTR POLL=0
```

The default is to omit the POLL parameter. Also, the LOAD LANZTR statement is commented out on systems that do not have a token ring adapter installed.

Activating Changes in the LANZ.NCF File

To activate the changes you make in the LANZ.NCF file:

- 1** Save the LANZ.NCF file.
- 2** Enter **ULANZ** at the server prompt to unload the agent.
- 3** Enter **LANZ** to reload the agent.

Using the Console Utility of the Traffic Analysis Agent for NetWare

The Traffic Analysis Agent for NetWare 1.3 provides a console utility (LANZCON.NLM) that performs the following three tasks:

- ◆ Enables or disables network monitoring by the selected network adapters
- ◆ Provides a source of detailed troubleshooting information
- ◆ Resolves a residual entry (for example, a Host TopN entry created by a management console that terminated unexpectedly)

When you install the Traffic Analysis Agent for NetWare, LANZCON.NLM is installed automatically in the SYS:\ZFS_AGNT\LANZ directory.

The following topics are discussed in greater detail in this section:

- ◆ [“Loading the Console Utility of the Traffic Analysis Agent for NetWare” on page 257](#)
- ◆ [“Enabling or Disabling Network Adapter Monitoring” on page 258](#)
- ◆ [“Viewing Network Adapter Information” on page 258](#)
- ◆ [“Viewing the Agent Item Status” on page 259](#)
- ◆ [“Accessing Detailed Information About Each Item” on page 260](#)
- ◆ [“Migrating Trend Files” on page 262](#)

Loading the Console Utility of the Traffic Analysis Agent for NetWare

To use LANZCON.NLM, enter the following command at the NetWare console prompt:

```
LOAD LANZCON CONTROLCOMMUNITY = <control community string>
```

IMPORTANT: If LANZCON is launched without any command line argument, then the default control community string is PUBLIC.

LANZCON.NLM is loaded and displays a list of network adapters, along with summary information about the network adapters currently installed on the server.

The following information is displayed for each network adapter:

- ◆ **Number (#):** The network adapter entry number in the network interface table.
- ◆ **Description:** A brief description of the network adapter.
- ◆ **Media Type:** The type of network connected to the network adapter: Ethernet, FDDI, or token ring.
- ◆ **Adapter Address:** The physical address of the network adapter.

Enabling or Disabling Network Adapter Monitoring

To enable or disable monitoring of a selected network adapter:

- 1** From the Network Adapters screen, select the appropriate adapter > press F3.
 - ◆ If the selected adapter is currently monitoring an Ethernet or token ring network, the console displays the Adapter Is Monitoring screen.
 - ◆ If the selected adapter is not monitoring an Ethernet or token ring network, the console displays the Adapter Is Not Monitoring screen.
- 2** Select Yes or No to enable or disable monitoring.

If you disable monitoring, all LAN analysis data for the selected adapter is deleted.

Using LANZCON, an FDDI adapter cannot be disabled. To disable an FDDI adapter:

- 1** Unload LANZCON, if loaded.
- 2** Unload LANZ, if loaded.
- 3** Open LANZ.NCF from SYS:\ZFS_AGNT\LANZ directory for editing.
- 4** Comment the statement LOAD LANZFDDI.NLM by entering the # symbol at the beginning of this statement.
- 5** Save LANZ.NCF and exit.
- 6** Reload LANZ.

Viewing Network Adapter Information

To bring up detailed information for network adapter items:

- 1** From the Network Adapters screen, select an adapter > press Enter.
- 2** From the Select Information to View screen, select Show Adapter Items.

The LANZCON utility displays the Network Adapter Items screen that lists all the items related to the selected network adapter.

The screen for a token ring adapter includes the information from the Novell Token Ring RMON MIB. For details, see [“Viewing the Agent Item Status” on page 259](#).

To return to the Select Information to View menu, press Esc.

The following information is provided for the selected adapter:

- ◆ **Item:** The types of items that are currently being monitored by the selected adapter. The Network Adapter Items screen shows a set of typical items consisting of token ring, Statistics, History, Host, Matrix, and Host TopN. The Traffic Analysis Agent for NetWare monitors these items by default. In the Network Adapter Items screen, the Host TopN item, indicating the list of the busiest nodes, has been added by a user. You can add other items to this display from ConsoleOne, depending on your configuration.

You can select any item to view more information about each topic. To view the values for the selected item, select the desired item > press Enter. Refer to the following sections for more examples of the screens.

- ◆ **Index:** The entry number of the displayed item in the list of all the items of the same type. The related tables are identified by this index.
- ◆ **Description:** A textual description of the entry. This column indicates the software entity or user that created the item. The items automatically monitored by the Traffic Analysis Agent for NetWare are indicated by the monitor.

For a token ring network entry, this column shows the media speed and the local ring number.

Viewing the Agent Item Status

When you click the Select Information to View menu > Show Agent Items, LANZCON displays all the items for each network adapter being monitored by the Traffic Analysis Agent for NetWare.

To view the agent item status for the selected agent:

- 1 From the Network Adapters screen, select an adapter > press Enter.
- 2 From the Select Information to View screen, select Show Agent Items.

The All NetWare LANalyzer Agent Items screen shows all the items related to the agent monitoring the segment. For example, if you are using multiple adapters to monitor multiple network segments, the screen lists all the items being monitored by the agent.

To delete any entry (except the token ring network entry), select the entry > click Delete > click Yes.

To return to the Network Adapter Items screen, press Esc.

The following information is provided for the agent:

- ◆ **Item:** The types of items available. The All NetWare LANalyzer Agent Items screen shows a set of typical items consisting of Statistics, History, Host, Matrix, and Host TopN. Additional items can be displayed, depending on your configuration.

You can select any item for more information about each topic. To view the values for an item, select the desired item > press Enter. See the following sections for more examples of the screens.

- ◆ **Index:** The entry number of the displayed item in the list of all items of the same type. The related tables are identified by this index.
- ◆ **Description:** A textual description of the entry. This column indicates the software entity or user that created the item table. The items automatically monitored by the Traffic Analysis Agent for NetWare are indicated by the monitor.

For a token ring network entry, this column shows the media speed and the local ring number.

Accessing Detailed Information About Each Item

This section describes the major categories of information available for both the selected network adapter and the Traffic Analysis Agent for NetWare. The following topics are covered:

- ◆ “Viewing the Token Ring RMON MIB Information” on page 260
- ◆ “Viewing the FDDI Ring RMON MIB Information” on page 260
- ◆ “Viewing Statistics Information” on page 260
- ◆ “Viewing History Information” on page 260
- ◆ “Viewing Host Information” on page 261
- ◆ “Viewing Matrix Information” on page 261

Viewing the Token Ring RMON MIB Information

To view the Token Ring RMON MIB information:

- 1** From the Network Adapter Items screen, select the token ring item > press Enter.
- 2** From the Select Information to View screen, select Show Adapter Items > press Enter.
- 3** Press Esc to exit this screen.

Viewing the FDDI Ring RMON MIB Information

To view the FDDI ring RMON MIB information:

- 1** From the Network Adapter Items screen, select the FDDI Ring item > press Enter.
- 2** From the Select Information to View screen, select Show Adapter Items > press Enter.

Viewing Statistics Information

The statistics information presents the basic statistics for each monitored adapter per segment.

To view the statistics information:

- 1** From the Network Adapter Items screen, select Statistics.
- 2** Press Enter.

For an Ethernet network entry, the LANZCON utility displays the Statistics Information screen.

This screen displays the statistical values of the selected network adapter. The display is updated periodically with the latest values for each field.

- 3** To exit this screen, press Esc.

Viewing History Information

The history information defines sampling functions for the networks that are being monitored. The History Control table defines a set of samples at a particular sampling interval for a particular network adapter.

To view the history information:

- 1** From the Network Adapter Items screen, select History > press Enter.
- 2** To exit this screen, press Esc.

The field descriptions are as follows:

- ◆ **Index:** An integer that uniquely identifies a row in the History Control table.
- ◆ **Data Source:** Identifies the network adapter and the Ethernet, FDDI, or token ring segment that is the source of the data for entries defined by this object.
- ◆ **Buckets Requested:** The requested number of discrete sampling intervals over which data will be saved in the portion of the media-specific table associated with this entry.
- ◆ **Buckets Granted:** The actual number of discrete sampling intervals over which data will be saved.
- ◆ **Interval:** The interval, in seconds, over which data is sampled for each bucket. The interval can be set to any number between 1 and 3,600 (one hour). The default interval for past hour is 30 seconds per sample, and the default interval for past day is 30 minutes (or 1,800 seconds) per sample.

The sampling scheme is determined by the buckets granted and the control interval.

- ◆ **Owner:** The entity that created the item. "Monitor" indicates that the item was created by the Traffic Analysis Agent for NetWare.
- ◆ **Status:** A status of Valid indicates that the agent is operating normally under the instructions given by the table.

Viewing Host Information

The host group gathers statistics about specific hosts or nodes on the LAN. The Traffic Analysis Agent for NetWare learns of new nodes on the LAN by observing the source and destination MAC addresses in good packets. For each node known to the agent, a set of statistics is maintained.

To view the host (node) information:

- 1 From the Network Adapter Items screen, select Host > press Enter.

The host group consists of three tables: two data tables and one control table. The two data tables are hostTable and hostTimeTable. The control table, hostControlTable, includes the following objects, which correspond to the fields displayed in the Host Information screen:

- ◆ **Index:** An integer that uniquely identifies a row in the hostControlTable. Each row in the control table refers to a unique network adapter, and thus, a unique segment.
- ◆ **Data Source:** Identifies the network adapter and the Ethernet, FDDI, or token ring segment that is the source of the data for the entries defined by this object.
- ◆ **Table Size:** The number of rows in the hostTable associated with this row.
- ◆ **Last Delete Time:** The value of the sysUpTime MIB object that corresponds to the last time an entry was deleted from the portion of the hostTable associated with this row. The value is zero (0) if no deletions occurred.
- ◆ **Owner:** Indicates the entity or user that created the item. "Monitor" indicates that the item was created by the Traffic Analysis Agent for NetWare.
- ◆ **Status:** A status of Valid indicates that the agent is operating normally under the instructions given by the table.

Viewing Matrix Information

The matrix group records information about the conversations between pairs of nodes on a network segment. The information is stored in the form of a matrix. This method of organization is useful to retrieve specific pairings of traffic information, such as finding out which nodes are making the most use of a server.

To view the matrix information:

- 1 From the Network Adapter Items screen, select Matrix > press Enter.

The matrix group consists of three tables: two data tables and one control table. The data tables are matrixSDTable and matrixDSTable. The control table, matrixControlTable, includes the following objects, which correspond to the fields displayed in the Matrix Information screen:

- ◆ **Index:** An integer that uniquely identifies a row in the matrixControlTable. Each row in the control table defines a function that discovers conversations on a particular network and places statistics about them in the two data tables.
- ◆ **Data Source:** Identifies the network adapter, and the Ethernet, FDDI, or token ring segment that are the source of the data for the entries defined by this object.
- ◆ **Table Size:** The number of rows in the matrixTable associated with this row.
- ◆ **Last Delete Time:** The value of the sysUpTime object that corresponds to the last time an entry was deleted from the portion of the matrixTable associated with this row. The value is zero (0) if no deletions occurred.
- ◆ **Owner:** Indicates the entity or user that created the item. "Monitor" indicates that the item was created by the Traffic Analysis Agent for NetWare.
- ◆ **Status:** A status of Valid indicates that the agent is operating normally under the instructions given by the table.

Migrating Trend Files

From ConsoleOne, you can view trends of traffic patterns on the monitored Ethernet, FDDI, and token ring segments. You can use the trend data to analyze traffic on the segment. For details, see [“Analyzing Trend Data for a Segment” on page 211](#).

Earlier versions of the Traffic Analysis Agent for NetWare (1.20 and 1.21) collected trend data that was sampled every one minute. The Traffic Analysis Agent for NetWare 1.30 that ships with ZfS collects trend data that are sampled every one minute, one hour, and one day. This functionality of version 1.30 of the Traffic Analysis Agent for NetWare ensures minimal communication between the agent and ConsoleOne, to reduce network traffic.

You can use the migrating tool (GTREND.EXE) to convert the trend data collected by earlier versions of the Traffic Analysis Agent for NetWare to trend data that can be used by version 1.30 of Traffic Analysis Agent for NetWare and ConsoleOne.

To migrate trend files collected by versions 1.20 or 1.21 of the Traffic Analysis Agent for NetWare:

- 1 Copy GTREND.EXE from the Installation CD to a TEMP folder on a 32-bit Windows NT, Windows 2000, Windows 95, or Windows 98 machine.
- 2 Copy the trend data files collected by earlier versions of the Traffic Analysis Agent for NetWare to the TEMP folder.
- 3 Run GTREND.EXE.

This will migrate the existing one-minute trend files to the corresponding one-hour and one-day trend files that can be used by version 1.30 of the Traffic Analysis Agent for NetWare.

- 4 Copy the migrated trend files to the SYS:\GTREND\ folder on the NetWare server and run the version 1.30 of the Traffic Analysis Agent for NetWare on the same server.

NOTE: The migration tool will not migrate older token ring trend data collected by version 1.20 or 1.21 of the Traffic Analysis Agent for NetWare because the older agents implemented a proprietary Token Ring MIB that enabled the agent to collect trend data sampled every one minute. Version 1.3 of the Traffic

Analysis Agent for NetWare implements the standard Token Ring MIB that supports historical trends (one minute, one hour and one day).

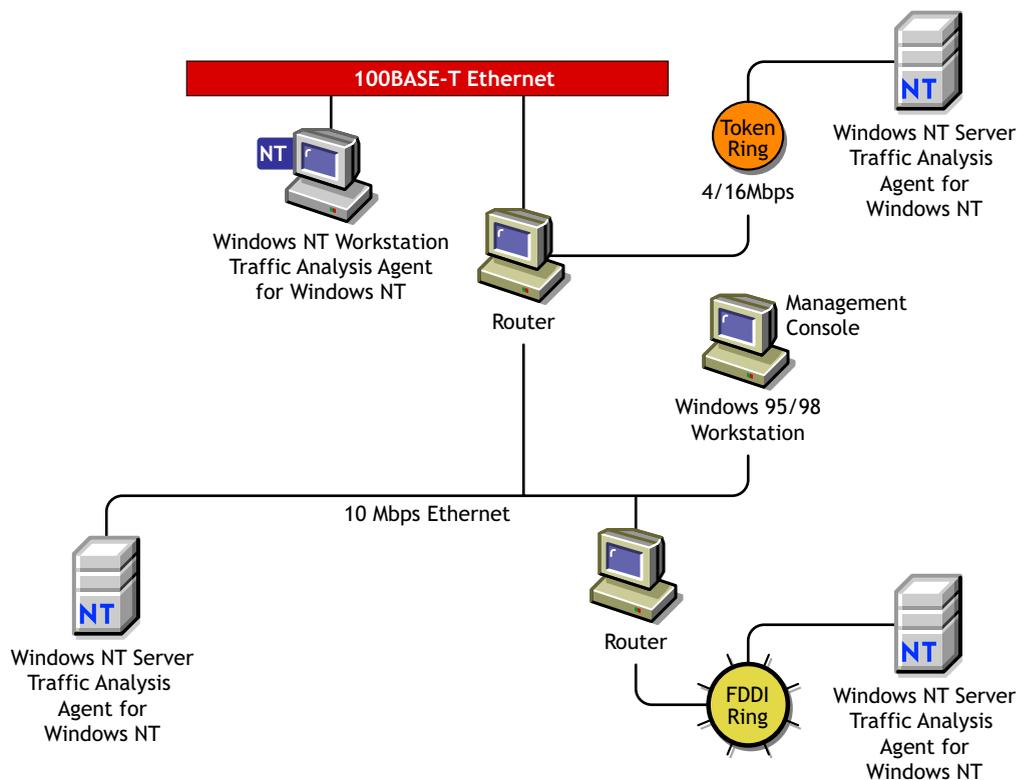
Using the Traffic Analysis Agent for Windows NT/2000

The Traffic Analysis Agent (version 1.30) for Windows NT/2000 runs on a Windows NT/2000 server or on a Windows NT workstation. The Traffic Analysis Agent for Windows NT/2000 monitors traffic on Ethernet, FDDI, or token ring segments.

The Traffic Analysis Agent for Windows NT/2000 is an RMON agent that implements functionality defined by the RMON MIB. It implements token ring extensions for RMON (RFC 1513 (<http://www.isi.edu/in-notes/rfc1513.txt>)) for token ring media, and a Novell proprietary MIB for FDDI media, in addition to implementing an RMON (RFC 1757 (<http://www.isi.edu/in-notes/rfc1757.txt>)) for Ethernet media.

The agent collects information about activity on your network and makes it available to ConsoleOne via SNMP. The Traffic Analysis Agent for Windows NT/2000 also implements the first two groups of RMON2 (RFC 2021 (<http://www.isi.edu/in-notes/rfc2021.txt>)).

The following figure illustrates a functional view of the Traffic Analysis Agent for Windows NT/2000:



Changes Made During Installation

When you install the Traffic Analysis Agent for Windows NT/2000, the following files are copied to Windows NT/2000:

Filename	Location	Description
LANZDIS.SYS	\\WINNT\\SYSTEM32\\DRIVERS	Kernel mode driver interface
LANZCTL.DLL	\\WINNT\\SYSTEM32	Control module
LANZMEM.DLL	\\WINNT\\SYSTEM32	Memory manager module
LANZLIB.DLL	\\WINNT\\SYSTEM32	Library module
LANZDI.DLL	\\WINNT\\SYSTEM32	User mode driver interface
LANZSM.DLL	\\WINNT\\SYSTEM32	Monitor module
LANZHIS.DLL	\\WINNT\\SYSTEM32	History module
LANZAEL.DLL	\\WINNT\\SYSTEM32	Alarm, event, and log module
LANZFCB.DLL	\\WINNT\\SYSTEM32	Filter capture, buffer module
LANZTR.DLL	\\WINNT\\SYSTEM32	Token ring manager module
LANZFDDI.DLL	\\WINNT\\SYSTEM32	FDDI manager module
GTREND.DLL	\\WINNT\\SYSTEM32	Trend module
LANZCON.EXE	\\LANZNT	Agent console application
LANZCON.HLP	\\LANZNT	Agent console help
LANZCON.CNT	\\ZFS_AGNT\\LANZCON	Required for online help from the application
GTREND.EXE	\\ZFS_AGNT\\LANZCON	Tool for migration of trend data from the older agent.
MGMTAPI.DLL	\\ZFS_AGNT\\LANZCON	SNMP application file
MSVCP50.DLL	\\ZFS_AGNT\\LANZCON	MFC APIs required for LANZCON
LANZCTL.DLL	\\ZFS_AGNT\\LANZCON	Required for LANZCON
MSFLXGRD.OCX	%SystemRoot%\\System32	Enables ActiveX* Controls in LANZCON

IMPORTANT: The default directory location for the LANZCON application is ZFS_AGNT\\LANZCON. You can change the location of LANZCON during installation.

The following sections provide information about optimizing and using the Traffic Analysis Agent for Windows NT/2000:

- ◆ [“Planning to Install the Traffic Analysis Agent for Windows NT/2000” on page 264](#)
- ◆ [“Optimizing the Traffic Analysis Agent for Windows NT/2000” on page 267](#)
- ◆ [“Using LANZCON” on page 270](#)

Planning to Install the Traffic Analysis Agent for Windows NT/2000

The Traffic Analysis Agent for Windows NT/2000 requires configuration of the Windows NT/2000 SNMP service before installing the agent. This section contains: Perform the following tasks to allow communication with the management server:

- ♦ WinNT
- ♦ Win2K

Installing and Configuring the Windows NT SNMP Service

Before installing the ZfS agent, you must install and configure the Windows NT SNMP service. This is required to enable communication with the management server.

To install and configure SNMP on Windows NT:

- 1** Install the SNMP service.
 - 1a** In the Control Panel, select Network > select Services > click Add.
 - 1b** Select SNMP Service from the Select Network Service dialog box.
 - 1c** Click OK.
 - 1d** Enter the full path to the Windows NT distribution files.
 - 1e** Click Continue.
- 2** Configure SNMP to start automatically.
 - 2a** In the Control Panel, double-click Services.
 - 2b** Click SNMP > Startup.
 - 2c** In the Startup Type options, select Automatic.
- 3** Configure the SNMP Trap service to start automatically.
 - 3a** In the Control Panel, double-click Services.
 - 3b** Click SNMP Trap Service > Startup.
 - 3c** In the Startup Type options, select Automatic.
- 4** Specify the trap community name and trap destination address so that the agent sends traps to the management server.
 - 4a** In the Control Panel, double-click Network.
 - 4b** Click the Services tab > select SNMP Service.
 - 4c** Click Properties.
 - 4d** Click the Traps tab.
 - 4e** Select a name from the Community Names box > click Add.
 The Add button is disabled if there are no Community Names available.
 - 4f** If the public community name is not present, type **public**.
 - 4g** Click Add.
 - 4h** Use the Trap Destinations box to add other DNS names and IP addresses in addition to the loopback IP address for the workstations or servers that should receive traps.
 - 4i** Click OK.
- 5** Set the SNMP security options trap community name so that SNMP packets from any host are accepted by the agent.
 - 5a** In the Control Panel, double-click Network.
 - 5b** Click the Services tab > select SNMP Service.

- 5c** Click Properties.
- 5d** Click the Security tab.
- 5e** In the Accepted Community Names box, click Add.
- 5f** In the Community Name box, type **public**.
The Accepted Community Names list displays the community names from which Windows NT will accept requests.
- 5g** Click Add.
- 5h** Select Accept SNMP Packets from Any Host > click OK.

Installing and Configuring the Windows 2000 SNMP Service

Before installing the ZfS agent, you must install and configure the Windows 2000 SNMP service. This is required to enable communication with the management server.

To install and configure SNMP on Windows 2000:

- 1** Install the SNMP service.
 - 1a** In the Control Panel, select Administrative Tools > Configure Your Server.
 - 1b** In the Application Server option, select Terminal Services.
 - 1c** Click Start.
 - 1d** In the Windows Components Wizard, double-click Management and Monitoring Tools.
 - 1e** Select Simple Network Management Protocol.
 - 1f** Click OK.
 - 1g** Click Next.
SNMP is started automatically after installation.
- 2** Configure the SNMP Trap service to start automatically.
 - 2a** In the Control Panel, select Administrative Tools > Services.
 - 2b** Click SNMP Trap Service > Startup.
 - 2c** In the Startup Type options, select Automatic.
- 3** Specify the trap community name and trap destination address so that the agent sends traps to the management server.
 - 3a** In the Control Panel, select Administrative Tools > Services
 - 3b** Double-click SNMP Service.
 - 3c** Click Properties.
 - 3d** Click the Traps tab.
 - 3e** Select a name from the Community Names box > click Add.
The Add button is disabled if there are no Community Names available.
 - 3f** If the public community name is not present, type **public**.
 - 3g** Click Add.
 - 3h** Use the Trap Destinations box to add other DNS names and IP addresses in addition to the loopback IP address for the workstations or servers that should receive traps.

- 3i** Click OK.
- 4** Set the SNMP security options trap community name so that SNMP packets from any host are accepted by the agent.
 - 4a** In the Control Panel, select Administrative Tools > Services.
 - 4b** Double-click SNMP Service.
 - 4c** Click Properties.
 - 4d** Click the Security tab.
 - 4e** In the Accepted Community Names box, click Add.
 - 4f** Select a name from the Community Name box.

The Accepted Community Names list displays the community names from which Windows 2000 will accept requests.
 - 4g** Click Add.
 - 4h** Select Accept SNMP Packets from Any Host > click OK.

IMPORTANT: After installing the SNMP services, you should re-install the service packs again.

Optimizing the Traffic Analysis Agent for Windows NT/2000

The Traffic Analysis Agent for Windows NT/2000 parameters are configured for optimal performance on Windows NT/2000. You can optimize the performance of the agent to suit your networking environment.

This section explains how to optimize the agent and monitor the functionality Traffic Analysis Agent for Windows NT/2000 using the agent console (LANZCON) for Windows NT/2000. For details, see [“Using LANZCON” on page 270](#).

The following sections explain the Traffic Analysis Agent for Windows NT/2000 configuration options:

- ◆ [“Configuring the Traffic Analysis Agent for Windows NT/2000” on page 267](#)
- ◆ [“Configuring the Modules of the Traffic Analysis Agent for Windows NT/2000” on page 268](#)
- ◆ [“Configuring the Parameters of the Traffic Analysis Agent for Windows NT/2000” on page 268](#)
- ◆ [“Automatically Loading the Agent with the SNMP Service” on page 269](#)

Configuring the Traffic Analysis Agent for Windows NT/2000

The Traffic Analysis Agent for Windows NT/2000 provides default values for modules and parameters. You can change the default values to optimize the performance of the Traffic Analysis Agent for Windows NT/2000.

You can configure the following modules of the Traffic Analysis Agent for Windows NT/2000:

- ◆ Packet Capture
- ◆ Station Monitor
- ◆ Token Ring Manager
- ◆ FDDI Manager

For details, see [“Configuring the Modules of the Traffic Analysis Agent for Windows NT/2000” on page 268.](#)

You can configure the following parameters of the Traffic Analysis Agent for Windows NT/2000:

- ◆ Memory Bound
- ◆ Memory Age
- ◆ Top N Station
- ◆ Generate Duplicate IP Address Alarms
- ◆ Trend Files Location

For details, see [“Configuring the Parameters of the Traffic Analysis Agent for Windows NT/2000” on page 268.](#)

Configuring the Modules of the Traffic Analysis Agent for Windows NT/2000

By default, all agent modules are enabled to load. You can choose to disable the modules.

To disable the modules of the Traffic Analysis Agent for Windows NT/2000:

- 1** From the LANZCON main menu, click **Configure > LANalyzer Agent Modules > Disable.**
- 2** Deselect the module you want the agent to monitor.
- 3** Click **OK.**

Configuring the Parameters of the Traffic Analysis Agent for Windows NT/2000

The Traffic Analysis Agent for Windows NT/2000 modules are loaded with default parameters. You can modify the parameters to optimize the performance of the agent.

The following table describes the parameters of the Memory Manager module:

Parameter	Default Value	Range	Description
Memory Bound	4 MB	1 MB - 10 MB	Sets the upper limit of available memory that can be allocated dynamically to the Traffic Analysis Agent for Windows NT/2000.
Memory Age	168 hours	1 hour - 720 hours	Controls the duration for which the Traffic Analysis Agent for Windows NT/2000 stores data in memory. When the duration setting is reached, existing data is purged from memory.

To modify the Memory Bound parameter:

- 1** From the LANZCON main menu, click **Configure > LANalyzer Agent Parameters.**
- 2** Click the **Memory Manager** tab.
- 3** Move the **Memory Bound** slider to the point you want to set as the memory bound value.

To modify the Memory Age parameter:

- 1** From the LANZCON main menu, click **Configure > LANalyzer Agent Parameters.**
- 2** Click the **Memory Manager** tab.

3 Move the Memory Age slider to the point you want to set as the memory age value.

IMPORTANT: Restart the Traffic Analysis Agent for Windows NT/2000 to ensure that the agent utilizes the changed parameter values. For details, see “[Installing and Setting Up Management and Monitoring Services](#)” in the *Installation* guide.

The following table describes the parameters of the Station Monitor module:

Parameter	Default Value	Range	Description
TopN Station	4 reports	2 - 10 reports	Controls the number of TopN reports the agent can generate.
Generate Duplicate IP Address Alarms	On	-	Controls the generation of duplicate IP address alarms.

To specify the number of TopN reports you want the agent to generate:

- 1** From the LANZCON main menu, click Configure > LANalyzer Agent Parameters.
- 2** Click the Station Monitor tab.
- 3** Select the number of TopN reports.

To stop generation of duplicate IP address alarms:

- 1** From the LANZCON main menu, click Configure > LANalyzer Agent Parameters.
- 2** Click the Station Monitor tab.
- 3** Deselect the Generate Duplicate IP Address Alarms check box.

The following table describes the Network Trend parameter:

Parameter	Default Path	Description
Trend Files Location	<i>system root</i> \GTREND	Specifies the directory path and location where trend files (*.GT) are created and updated.

IMPORTANT: If you delete the *.GT file, all the previous trend information will be lost.

To specify a path to a location for storing trend data:

- 1** From the LANZCON main menu, click Configure > LANalyzer Agent Parameters.
- 2** Click the Network Trends tab.
- 3** Enter or browse to select the directory path to the location where you want the Traffic Analysis Agent for Windows NT/2000 to store trend data.

Automatically Loading the Agent with the SNMP Service

The Traffic Analysis Agent depends on the Microsoft* SNMP service on Windows NT/2000. When SNMP starts, it loads agent DLLs in its address space. Once the agent is installed, it will be always loaded by the SNMP service, by default, whenever the service starts.

You can enable or disable loading of the agent DLLs with SNMP by checking the desired options in the Novell Traffic Analysis Agent Loading with SNMP dialog box. If you disable the agent, the SNMP service will start normally but the Traffic Analysis Agent will not work. The Traffic

Analysis Agent will neither capture packets by placing the NIC cards into the promiscuous mode nor will respond to SNMP requests.

Using LANZCON

This section explains how you can use the LANZCON utility to configure and diagnose the Traffic Analysis Agent for Windows NT/2000.

LANZCON for Windows NT/2000 is a graphical user interface provided by the Traffic Analysis Agent for Windows NT/2000 to configure the agent modules and parameters and to diagnose the agent. You can use LANZCON to obtain information about network segments monitored by the agent to help you troubleshoot problems.

To open the LANZCON utility, do one of the following:

- ◆ From the Windows NT/2000 Programs menu, click LANalyzer Agent for Windows NT > LANZCON.
- ◆ Double-click the LANZCON icon  on your desktop.

You can perform the following tasks with LANZCON:

- ◆ “Viewing Network Adapters” on page 270
- ◆ “Enabling or Disabling Network Adapter Monitoring” on page 271
- ◆ “Viewing the Agent Log” on page 271
- ◆ “Viewing the Agent Status” on page 271
- ◆ “Viewing RMON Tables” on page 271
- ◆ “Viewing SNMP Traps” on page 272

Viewing Network Adapters

On loading LANZCON, you will see the Network Adapters window. The Network Adapters window displays information about monitored adapters in two panes.

The following table describes the two panes in the Network Adapters window:

Pane	Displays	Description
Left pane	Adapter Tree view	Displays a list of network adapters discovered by the Traffic Analysis Agent for Windows NT/2000. The default view displays a collapsed tree. You can expand each network adapter in the tree to view the list of RMON tables for the selected adapter.
Right pane	Table view	Displays details about the object you select in the left pane. If you select an adapter in the left pane, interface table (RFC 1213 (http://www.isi.edu/in-notes/rfc1213.txt)) details such as media type, MAC address, and description of the selected adapter are displayed in the right pane. If you select an RMON table in the left pane, table data is displayed in the right pane.

Enabling or Disabling Network Adapter Monitoring

The Traffic Analysis Agent for Windows NT/2000 collects information about monitored adapters and displays it in the right pane of the Network Adapters window.

By default, adapter monitoring is enabled. LANZCON lets you disable adapter monitoring. If you disable adapter monitoring, the Traffic Analysis Agent for Windows NT/2000 will stop collecting data for the adapter and the RMON tables for the adapter will be deleted.

IMPORTANT: You cannot disable monitoring FDDI adapters through LANZCON.

To enable adapter monitoring:

- 1 Select an adapter in the left pane of the Network Adapters window.
- 2 Click View > NetWork Adapters > Enable.

To disable adapter monitoring:

- 1 Select an adapter in the left pane of the Network Adapters window.
- 2 Click View > NetWork Adapters > Disable.

Viewing the Agent Log

The Traffic Analysis Agent for Windows NT/2000 logs significant events and error messages that occurred during a session.

To view the agent log:

- 1 From the LANZCON main menu, click View > Agent Log.

Viewing the Agent Status

You can view the status of the agent from the LANalyzer Agent Status window. The agent status window indicates whether the agent modules are loaded or not loaded.

To view the agent status:

- 1 From the LANZCON main menu, click View > Agent Status.

Viewing RMON Tables

RMON tables are listed under each network adapter. You can view the RMON tables by selecting a table in the left pane of the Network Adapters window. RMON table data is displayed in the right pane.

The Network Adapter tree displays the following RMON tables:

- ◆ Statistics
- ◆ History Control
- ◆ History Data
- ◆ Host Control
- ◆ Host Entry
- ◆ Host TopN Control
- ◆ Host TopN Entry
- ◆ Matrix Control

- ◆ Matrix SD Entry
- ◆ Filter, Channel, and Buffer

The Alarm Information tree displays the following RMON tables:

- ◆ Alarm
- ◆ Event
- ◆ Log

Viewing SNMP Traps

The Traffic Analysis Agent for Windows NT/2000 monitors network segments and sends traps to the management server. ConsoleOne displays the alarm when it receives the trap from the management server.

Trap information is displayed in the SNMP Traps window. For each trap, the table shows trap data that can be obtained.

Statistic	Explanation
Receive Time	Displays the time when the trap occurred
Trap Summary	Displays a description of the trap

IMPORTANT: LANZCON will receive trap notifications if you have ensured that Windows NT/2000 SNMP has been configured to send traps to a loopback trap destination address. For details, see ["Planning to Install the Traffic Analysis Agent for Windows NT/2000" on page 264](#).

To view SNMP traps from LANZCON main menu, click > View > SNMP Traps.

9

Customizing Agent Configuration

The ZENworks[®] for Servers (ZfS) server management SNMP agents run on NetWare[®] and Windows* NT* servers in your network. The agents monitor servers, collecting historical data and dynamic data in response to requests from ConsoleOne[®]. An administrator at the ZfS ConsoleOne can request data simply by clicking a representative icon for any device, operating system, or service discovered on a server.

After the Management Agent for NetWare and the Management Agent for Windows NT have been installed on your network NetWare and Windows NT servers, they are ready to operate with the default settings. In most cases, this configuration is sufficient; however, you can customize the agent settings to enhance management functionality.

This appendix contains the following sections:

- ◆ [“Agent Files” on page 273](#)
- ◆ [“Customizing the Management Agent for NetWare” on page 275](#)
- ◆ [“Customizing the Management Agent for Windows NT Server” on page 278](#)

Agent Files

The following sections describe the agent files that are installed on each managed server:

- ◆ [“Management Agent for NetWare Files” on page 273](#)
- ◆ [“Management Agent for Windows NT Server Files” on page 274](#)

Management Agent for NetWare Files

The following table describes the Management Agent for NetWare NLM[™] files installed on a NetWare server:

Management Agent for NetWare NLM Files	Description
SERVINST.NLM	Implements the NetWare server MIB (NWSERVER.MIB).
HOSTMIB.NLM	Implements the standard Host Resources MIB [RFC 1514] and Novell [®] extensions to that MIB (NWHOSTX.MIB).
NTREND.NLM	Implements the Threshold and Trend MIB (NWTREND.MIB). When loaded, NTREND.NLM sets trends and thresholds for each monitored attribute according to the server's configuration. The NTREND.INI file contains configuration parameters for NTREND.NLM.
NWTRAP.NLM	Implements the NetWare Server Trap MIB (NWLARM.MIB). The NWTRAP.CFG file contains configuration parameters for NWTRAP.NLM.

Management Agent for NetWare NLM Files	Description
FINDNMS.NLM	Used by NetWare servers running the Management Agent for NetWare. Employ FINDNMS.NLM to listen for SNMP Management console advertising themselves using the Service Advertising Protocol (SAP) number 0x026a. FINDNMS.NLM then adds the Internetwork Packet Exchange™ (IPX™) address of each ConsoleOne discovered to the list of stations that receive traps.
NDSTRAP.NLM	Implements the NDSTRAP.MIB to capture and forward Novell eDirectory events to SNMP Management console.
MONDATA.NLM	Allows you to monitor NetWare servers.

The following table provides a brief description of the enterprise MIBs associated with the Management Agent for NetWare:

MIB Name	Description
NDSTRAP.MIB	A Novell proprietary MIB designed to capture eDirectory events and forward them to SNMP Management console as SNMP traps. There are more than 130 traps currently in the MIB and new ones are being added as they are identified.
NWALARM.MIB	A Novell proprietary MIB that handles all the NetWare Core OS alerts and forwards them as SNMP traps. It currently supports more than 375 traps and new ones are being added as they are identified.
NWHOSTX.MIB	A Novell extension to RFC1514 (the Host Resources MIB). It adds devices and components that are specific to NetWare that were not directly included in RFC1514.
NWSERVER.MIB	A Novell proprietary MIB that is the basis for NetWare Core OS management. More than 300 objects are identified in this MIB. Access to the parameters that can be set from the console for both GET and SET is defined. The MIB has several groups and tables for users, file systems, volumes, queues, Open Data-Link Interface™ (ODI™), set parameters, and so forth.
NWTMSYNC.MIB	A Novell proprietary MIB that allows for SNMP management of TIMESYNC.NLM. It provides access to the list of time sources as well as time clients. You may also access the clock structure through this MIB.
NWTREND.MIB	A Novell proprietary MIB that keeps track of objects that are most useful when tracked over a period of time. For example, CPU utilization and packets received have limited value as static numbers, but when monitored at regular intervals for a period of time, they tell a great deal about what is happening on a server. This MIB also lets you set user-definable thresholds for the managed objects and will send SNMP traps when a threshold is exceeded.
RFC1514.MIB	The Internet Standard Host Resources MIB. It defines general categories about a host machine, including physical components of the system such as disks, memory, CPU, printers, adapter cards, and so forth.

Management Agent for Windows NT Server Files

Following is a list of files that can be manually configured with a text editor to modify default results of the Management Agent for Windows NT:

Management Agent for Windows NT Server .INI Files	Description
N_NTTREN.INI	Specifies the initial values for the trends and thresholds supported by the Management Agent for Windows NT.
NTTRAP.INI	Specifies settings to troubleshoot your Windows NT server that runs the Management Agent for Windows NT and settings to enable you to send Windows NT events to the management system as SNMP traps.
NTHOST.INI	Specifies the SNMP settings supported by the Management Agent for Windows NT.
N_NTFMW.INI	Allows you to specify IPX addresses that will be ignored and will not receive SNMP traps.

The following table provides a brief description of the enterprise MIBs associated with the Management Agent for Windows NT. In addition, the Management Agent for Windows NT converts all Windows NT system, security, and application events to SNMP traps.

MIB Name	Description
NTSERVER.MIB	Gives minimal Windows NT system information like Server Name, OS, major and minor versions, time zone, remote and local volumes count, etc.
RFC1514.MIB	The Internet Standard Host Resources MIB. It defines general categories about a host machine, including physical components of the system such as disks, memory, CPU, printers, adapter cards, and so forth.
NTTRAP.MIB	A generic MIB based on RFC1514. Windows NT events that are converted into traps are forwarded to the ZfS network management system.
NTTREND.MIB	A Novell proprietary MIB that keeps track of objects that are most useful when tracked over a period of time. For example, CPU utilization and packets received have limited value as static numbers, but when monitored at regular intervals for a period of time, they tell a great deal about what is happening on a server. This MIB also lets you set user-definable thresholds for the managed objects and will send SNMP traps when a threshold is exceeded.

Customizing the Management Agent for NetWare

The Management Agent for NetWare installation process creates the NMA2.NCF file (NetWare 3.x and 4.x servers) or the NMA5.NCF file (NetWare 5.x servers) in the SYS:\ZFS_AGNT\NMA directory. When the NetWare server is started, this file automatically loads all the NLM files required for the Management Agent for NetWare in a default configuration state. There are, however, several LOAD parameters that you can configure for each of the NLM files used with the agent.

You can configure your server to use these options by editing the NMA2.NCF or NMA5.NCF file on your server. Also, if your server is already running, you can unload any of these NLM files and then load them at the NetWare server console using any of the configuration parameters. You can configure these parameters at the NetWare server console or by using the NetWare remote console utility, RCONSOLEJ.

The sections that follow describe each of the command-line parameters that you can configure for the Management Agent for NetWare.

- ◆ “SERVINST.NLM Load Parameters” on page 276
- ◆ “HOSTMIB.NLM Load Parameters” on page 276
- ◆ “NTREND.NLM Load Parameters” on page 277

SERVINST.NLM Load Parameters

SERVINST.NLM implements the NWSERVER.MIB NetWare Server MIB. You can load SERVINST.NLM at the command line with any or all of the following parameters:

```
LOAD SERVINST D, U=n, V, B=n H
```

Parameter	Description
D	DisableSets: If this parameter is present, SERVINST.NLM does not allow SNMP SET commands for objects in NWSERVER.MIB. Default: SETS enabled (subject to SNMP security).
U= <i>n</i>	UpdateInterval= <i>n</i> : Sets the list update interval to <i>n</i> (<i>n</i> is a value in seconds). This determines how often certain internal lists kept by SERVINST.NLM (such as volumes and queues) are updated. Set this parameter higher to minimize the number of CPU cycles used by SERVINST.NLM, or lower to guarantee immediate reporting of server status changes that affect the lists. Default: 300 seconds.
V	Verbose: Displays informational messages. Default: Off.
B= <i>n</i>	BuildUserListHour= <i>n</i> : The local time each day on a 24-hour clock (0 to 23) at which the SERVINST.NLM software builds a list of users that have access to the server. Default: 2 (2:00 AM).
H	Help: Displays help on command line parameters. If you use the H parameter, SERVINST.NLM displays the help messages and then exits. It does not remain loaded even if other parameters are entered on the command line. Default: Off.

HOSTMIB.NLM Load Parameters

HOSTMIB.NLM implements both the standard Host Resources MIB (RFC 1514) and the Novell extensions to the Host Resources MIB (NWHOSTX.MIB). You can load HOSTMIB.NLM at the command line with any or all of the following parameters:

```
LOAD HOSTMIB.NLM D, U=n, V, H
```

Parameter	Description
D	<p>DisableSets: If this parameter is present, HOSTMIB.NLM does not allow SNMP SET commands for objects in RFC1514.MIB or NWHOSTX.MIB.</p> <p>Default: SETS enabled (subject to SNMP security).</p>
U= <i>n</i>	<p>UpdateInterval=<i>n</i>: Sets the list update interval to <i>n</i> (<i>n</i> is a value in seconds). This determines how often certain internal lists kept by HOSTMIB.NLM are updated. Set this parameter higher to minimize the number of CPU cycles used by HOSTMIB.NLM, or lower to guarantee immediate reporting of server status changes that affect the lists.</p> <p>Default: 60 seconds.</p>
V	<p>Verbose: Displays informational messages.</p> <p>Default: Off.</p>
H	<p>Help: Displays help on command-line parameters. If you use the H parameter, HOSTMIB.NLM displays the help messages and then exits. It does not remain loaded even if other parameters are entered on the command line.</p> <p>Default: Off.</p>

NTREND.NLM Load Parameters

NTREND.NLM implements the Threshold and Trend MIB (NWTREND.MIB).

When first loaded, NTREND.NLM automatically sets trends and thresholds for each monitored attribute according to the server's configuration from values stored in the NTREND.INI file (located in the SYS:\ETC directory). You can edit this file as described in [“Setting Default Trends and Thresholds” on page 130](#).

Thereafter, as configuration changes occur over time, NTREND.NLM adjusts to changes in the number and type of physical network interfaces, queues, volumes, and disks. Default thresholds are set only for important parameters. You can later use SNMP SET commands to set thresholds for parameters such as files read and packets in.

A trend file is created for each monitored attribute instance, even if trending is disabled for that object. The file header contains all the information from nwtControlTableEntry, and the rest of the file stores the sample history (if any). Once a trend file is created, it exists until explicitly deleted by the operator, even if the monitored object (a queue, for example) no longer exists. When a monitored object no longer exists, the associated nwtControlStatus is recorded as invalid.

You can load NTREND.NLM at the command line with any or all of the following parameters:

```
LOAD NTREND D=dir, R, V, H
```

Parameter	Description
D= <i>dir</i>	<p>Directory=<i>dir</i>: Enables you to specify the volume and directory where NTREND.NLM stores the history data files. Example: To use VOL1:\TEST as the directory for trending files, enter the following command:</p> <pre>load ntrend D=vol1:\test</pre> <p>Default: SYS:\NTREND.</p>

Parameter	Description
R	Reset: Causes NTREND.NLM to discard all the old trending history data and restart the sampling.
V	Verbose: Displays informational messages. Default: Off.
H	Help: Displays help on command-line parameters. Default: Off.

Customizing the Management Agent for Windows NT Server

You can manually edit the following files to modify the default Management Agent for Windows NT configuration on a managed Windows NT server.

Management Agent for Windows NT Server .INI Files	Description
N_NTFMW.INI	Allows you to specify IPX addresses that will be ignored and will not receive SNMP traps.
NTTRAP.INI	Specifies settings to troubleshoot your managed Windows NT servers and set trap filters to specify which Windows NT events are sent to the management system as SNMP traps. See “Controlling Alarm Generation” on page 134 for detailed information on configuring trap filters and trap generation.
N_NTTREN.INI	Specifies the initial values for the trends and thresholds supported by the Management Agent for Windows NT. See “Setting Default Trends and Thresholds” on page 130 for detailed information on modifying default trends and thresholds.

Configuring the Management Agent for Windows NT Server

By default, the Management Agent for Windows NT sends traps to SNMP Management console on IPX networks broadcasting the 0x026 Service Advertising Protocol (SAP) ID. You can edit the N_NTFMW.INI file to include the IPX addresses of SNMP Management console that you do not want to include as trap targets.

To add the IPX address of a SNMP Management console to omit as an automatic trap recipient:

- 1 Open the N_NTFMW.INI file in a text editor.
- 2 Add the IPX address for omitted SNMP Management console using the following syntax:

```
xxxxxxx.yyyyyyyyyyy
```

where xxxxxxxx is the net address and yyyyyyyyyyy is the node address, such as 01014044.00001B4DDAFD.

- 3 Save the file and restart the Management Agent for Windows NT.

Third-Party Agent Configuration

Third-party SNMP agents require the following tasks to be completed before traps are received:

- ♦ [“Ensuring that Traps Are Received” on page 279](#)
- ♦ [“Integrating Vendor-Specific SNMP Traps” on page 279](#)

Ensuring that Traps Are Received

When configuring the SNMP agent or SNMP Remote Network Monitoring (RMON) agent on a network device, configure the agent’s trap destination list (trap-target list) to include the ZfS management server station IP address or server name. Refer to the agent’s documentation for information on configuring this. ConsoleOne displays alarms for all devices that forward alarms to the management server.

If your network device is using the Management Agent for NetWare, Management Agent for Windows NT Server, NetWare LANalyzer[®] Agent[™], or the LANalyzer Agent for Windows NT, the agent’s trap destination list is automatically configured for you. For information on configuring the trap destination list, see [“Setting Up Discovery” on page 88](#) for configuration information and [Chapter 8, “Understanding Traffic Analysis,” on page 191](#) for information on configuring the RMON agents.

Integrating Vendor-Specific SNMP Traps

Before AMS can process the alarm, you must include vendor-specific MIBs for the third-party SNMP agents in the management server MIB pool. You can further integrate third-party SNMP agents by annotating the trap definitions in the vendor MIB.

AMS interprets ASN.1 annotations to trap definitions in a MIB to set the severity level and device status assigned to an alarm. The MIBs included with ZfS already include the proper annotations. The annotations provide detail on severity levels and device status to the AMS.

See [Chapter 6, “Using the MIB Tools,” on page 165](#) for information on adding a MIB to the management server’s MIB pool and annotating third-party MIBs.

ZENworks[®] for Servers (ZfS) provides packet capture and decoding tools that help you analyze the network activity and identify the source of network problems. Capturing and decoding packets can help you troubleshoot network problems by giving you detailed information about segment activity. For details, see [“Capturing Packets” on page 223](#) and [“Displaying Captured Packets” on page 226](#).

This section provides information about decoding support provided by ZfS for the following protocol suites:

- ◆ [“NetWare Protocol Suite” on page 281](#)
- ◆ [“Network File System Protocol Suite” on page 283](#)
- ◆ [“Systems Network Architecture Protocol Suite” on page 283](#)
- ◆ [“AppleTalk Protocol Suite” on page 283](#)
- ◆ [“TCP/IP Protocol Suite” on page 285](#)

NetWare Protocol Suite

NetWare[®] contains a group of protocols that perform various functions in a NetWare network. Each protocol in the NetWare protocol suite works with the IPX[™] protocol. ZfS supports the following protocols in the NetWare suite of protocols:

NetWare Protocol	Description
BCAST	NetWare Broadcast Message Notification. The protocol a NetWare server uses to inform an idle workstation that a message is pending. This message appears on the top or bottom line of the monitor on DOS stations.
DIAG	Diagnostic Responder. A protocol used for connectivity testing and information gathering. By default, NetWare clients use the Diagnostic Responder to reply to diagnostic requests.
IPX	Internetwork Packet Exchange [™] . A protocol that routes outgoing data packets across a network. Every NetWare network has a unique address assigned when its servers are configured. IPX routers use this address to route packets through an internetwork. IPX makes routing decisions based on information compiled by the Routing Information Protocol (RIP).
LSP	NetWare Lite [™] Sideband Protocol. A connectionless (datagram) oriented protocol that operates as a sideband for NetWare Lite Transport Protocol (NLTP) connections.
NBIOS	NetBIOS. An emulator that allows workstations to run applications that support IBM* NetBIOS calls. NetBIOS is the IBM standard protocol for applications developed to run peer-to-peer communications on token ring networks.

NetWare Protocol	Description
NCP™	<p>NetWare Core Protocol™. A set of procedures that a file server operating system follows to accept and respond to workstation requests.</p> <p>An NCP exist for every service a workstation might request from a file server. Common requests handled by the NCP protocols include creating or deleting a file, manipulating directories and files, performing a directory listing, altering the bindery (drive mappings and security), and printing.</p>
NDS®	<p>The NDS protocol, called the Novell Directory Access Protocol (NDAP), is a wire protocol that allows Novell eDirectory to service client requests and to send client requests to other Novell eDirectory servers. NDAP is built based on NCP.</p>
NLP	<p>NetWare Lite Protocol. A protocol that is an integral part of NetWare Lite, which operates on top of the Novell IPX protocol. NLP is an application-layer and service-layer protocol that performs file system and print functions. NLP also uses NLTP, which is similar in function to the transport protocol used in NCP.</p>
NLSP™	<p>NetWare Link Services Protocol™. A link-state routing protocol designed for IPX internetworks.</p>
RIP	<p>Routing Information Protocol. A protocol that automates the process of updating routing tables. Routing is the process of moving network packets between separate networks. With RIP, when one router learns about changes in its routes, it broadcasts this information to neighboring routers so they can update their routing tables. As a result, if a network component fails (such as a router or a phone line), the other network components can inform each other of alternate routes. When the faulty component is repaired, the network changes back to the previous condition.</p>
SAP	<p>Service Advertising Protocol. A protocol that lets NetWare servers advertise their services by name and type. A workstation can broadcast a request to find all services available or a specific service closest to the client.</p>
SER	<p>Novell Serialization (Copy Protection) Packets. Packets that NetWare servers send to other NetWare servers to ensure that each server has a unique serial number.</p>
SNMP	<p>Simple Network Management Protocol. An application-layer protocol designed to facilitate the exchange of management information between network devices. By using SNMP to access management information data (such as packets per second and network error rates), network administrators can easily manage network performance and find and solve network problems.</p>
SPX™	<p>Sequenced Packet Exchange™. A connection-oriented transport protocol that monitors network transmissions to ensure successful delivery of packets. SPX enhances the IPX protocol by supervising data sent across the network. SPX can track data transmissions consisting of a series of separate packets.</p> <p>SPX also requests acknowledgments from and returns acknowledgments to a communications partner, ensuring successful data delivery. If an acknowledgment request brings no response within a specified time, SPX retransmits the request. After a reasonable number of retransmissions fail to return a positive acknowledgment, SPX assumes the connection has failed and reports the error.</p> <p>The NetWare print server uses SPX.</p>
WDOG	<p>Watchdog. A maintenance protocol provided with NetWare. Watchdog monitors stations that are logged in to a NetWare server. Watchdog determines whether the NetWare shells are still operating and, if not, releases the connection.</p>

Network File System Protocol Suite

The Network File System (NFS) suite of protocols is described in the following table.

Network File System Protocol	Description
MOUNT	The MOUNT protocol, used in conjunction with NFS, performs operating system-specific functions that allow NFS clients to attach remote directory trees to a point within the local file system.
NFS	Network File System. This protocol provides transparent remote access to shared file systems across networks. NFS uses Remote Procedure Call (RPC) and is machine, operating system, network architecture, and transport protocol independent.
PORTMAP	The PORTMAP protocol converts RPC program numbers into Transmission Control Protocol/User Datagram Protocol (TCP/UDP) port numbers. When a client wants to make an RPC call to a given program number, it will first contact PORTMAP on the remote machine to determine the port number where RPC packets should be sent.
RPC	Remote Procedure Call. This protocol allows a program on one computer to make a subroutine call on a remote computer. Every subroutine or remote procedure is identified by a unique program number.

Systems Network Architecture Protocol Suite

The Systems Network Architecture (SNA) suite of protocols is described in the following table.

Systems Network Architecture Protocol	Description
RH	Request/Response Header. This protocol carries the SNA Request/Response Units as its payload.
RU	Request/Response Unit. An SNA client uses this protocol to communicate with an SNA server.
TH	Transmission Header. This protocol runs on a data link layer and serves as the transmission layer for an SNA Path Information Unit.
XID	Exchange Station Identification. An SNA node uses this protocol to check whether its peer SNA node is ready for communication and to exchange its station details with it.

AppleTalk Protocol Suite

The AppleTalk* and AppleTalk-related suite of protocols is described in the following table.

AppleTalk Protocol	Description
AARP	AppleTalk Address Resolution Protocol. An AppleTalk protocol that reconciles addressing differences between a data link protocol and the rest of a protocol family. For example, by resolving the differences between an Ethernet addressing scheme and the AppleTalk addressing scheme, AARP facilitates the transport of datagram delivery protocol (DDP) packets over a high-speed EtherTalk* connection.
ADSP	AppleTalk Data Stream Protocol. A connection-oriented protocol that provides a reliable, full-duplex, byte stream service between any two sockets in an AppleTalk internetwork. ADSP ensures sequential, duplicate-free delivery of data over its connections.
AEP	AppleTalk Echo Protocol. A simple protocol that allows a node to send a packet to any other node in an AppleTalk internetwork and receive an echoed copy of that packet in return.
AFP	AppleTalk Filing Protocol. A presentation layer protocol that allows users to share data files and applications that reside in an AppleTalk shared resource, such as a file server.
ASP	AppleTalk Session Protocol. A general, all-purpose protocol that uses the services of the AppleTalk Transaction Protocol (ATP) to provide session establishment, maintenance, and tear-down, along with request sequencing.
ATP	AppleTalk Transaction Protocol. A transport protocol that provides a loss-free transaction service between sockets. This service allows exchanges between two socket clients in which one client requests the other to perform a particular task and report the results. ATP binds the request and response together to ensure the reliable exchange of request-response pairs.
E-DDP	Extended Datagram Delivery Protocol. A datagram delivery protocol that uses an extended header. An extended header is required for packets that are transmitted from one network to another network within an AppleTalk Internet.
ELAP	EtherTalk Link Access Protocol. The link-access protocol used in an EtherTalk network. It is built on the top of the standard Ethernet data link layer.
NBP	Name Binding Protocol. A transport layer protocol that translates a character string name into the internetwork address of the corresponding socket client. NBP enables AppleTalk protocols to understand user-defined zones and device names by providing and maintaining translation tables that map these names to corresponding socket addresses.
PAP	Printer Access Protocol. This protocol manages interaction between workstations and print servers. It handles connection setup, maintenance, and termination. It can also handle data transfer.
RTMP	Routing Table Maintenance Protocol. This AppleTalk protocol establishes and maintains the routing information that is required by internetwork routers to route datagrams from any source socket to any destination socket in the internetwork. Using RTMP, internetwork routers dynamically maintain routing tables to reflect changes in internetwork topology.
S-DDP	Short Datagram Delivery Protocol. A DDP that uses a short header. A short header is often used for packets whose source and destination sockets are within the boundaries of a single AppleTalk network.
ZIP	Zone Information Protocol. A protocol that maintains up-to-date routing information across the internetwork.

TCP/IP Protocol Suite

The TCP/IP suite of protocols is described in the following table.

TCP/IP Protocol	Description
ARP	<p>Address Resolution Protocol. A protocol used by a host to determine the hardware address of another host. A TCP/IP system contains a table that maps IP addresses to the hardware addresses of the different hosts and routers on the internetwork. This table works in much the same way as a host table, translating an IP address to an Ethernet address. Unlike the host table, however, the ARP table is not usually maintained by you or your network administrator. The ARP protocol creates entries in this table as needed.</p> <p>If the hardware address of the destination is not found in your station's ARP table, a broadcast is sent to every host on the network requesting the address. If that host is up and supports the ARP protocol, it receives the broadcast from your station and responds by sending its hardware address back to your station. This address is then added to your station's ARP table.</p>
IMAP	<p>IMAP stands for Internet Message Access Protocol. It is a method of accessing electronic mail or bulletin board messages that are placed on a (possibly shared) mail server. It permits a "client" e-mail program to access remote message stores as if they were local.</p>
BOOTP	<p>BootStrap Protocol. This protocol allows a diskless workstation to determine its IP address and other information without using the Reverse Address Resolution Protocol (RARP).</p>
DHCP	<p>Dynamic Host Configuration Protocol. This protocol supplies hosts with configuration parameters, leases dynamically allocated IP addresses, and acts as an enhancement to BOOTP.</p>
DNS	<p>Domain Name System. The distributed naming service used on the Internet. DNS provides a computer's IP address if domain names exist for the computer.</p>
FTP	<p>File Transfer Protocol. TCP/IP application-layer protocol that supports file transfers.</p>
HTTP	<p>Hypertext Transfer Protocol. An application-layer protocol that Web browsers and Web servers use to communicate with each other.</p>
ICMP	<p>Internet Control Message Protocol. A protocol that works with IP to provide routing efficiency and error information. ICMP is part of the TCP/IP protocol suite. Because IP is connectionless, it cannot detect anomalous internetwork conditions. ICMP works with IP to provide TCP or other upper-layer protocols with this information.</p>
IGMP	<p>Internet Group Management Protocol. A protocol used by IP hosts to report their multicast group memberships to routers. The protocol is also used to query routers on memberships and to generate reports on group membership. Termination of group membership can be quickly reported using this protocol.</p>

TCP/IP Protocol	Description
IP	<p>Internet Protocol. A protocol that provides connectionless, nonguaranteed delivery of transport layer packets (also called transport protocol data units or TPDUs) across an internetwork. IP is part of the TCP/IP protocol suite.</p> <p>IP can fragment TPDUs into smaller parts, if necessary, and then reassemble them at an intermediate station (usually a router) or at their destination host.</p> <p>Each TPDU or fragment is fitted with an IP header and transmitted as a packet by lower-layer protocols. IP moves datagrams through the internetwork, one hop at a time. If a TPDU fragment arrives at its destination out of order, IP reassembles the fragments, in sequence, at the destination.</p>
LDAP	<p>Lightweight Directory Access Protocol. This protocol provides access to the x.500 Directory while not incurring the resource requirements of the Directory Access Protocol (DAP). LDAP is specifically targeted at simple management applications and browser applications that provide read/write interactive access to the x.500 Directory, and is intended to be a complement to the DAP itself.</p>
NFS	<p>The Network File System (NFS) protocol provides transparent remote access to shared files across networks. The NFS protocol is designed to be portable across different machines, operating systems, network architectures, and transport protocols. This portability is achieved through the use of Remote Procedure Call (RPC) primitives built on top of an eXternal Data Representation (XDR).</p>
NTP	<p>Network Time Protocol. A protocol used to synchronize timekeeping among a set of distributed time servers and clients. It is used to convey timekeeping information in a hierarchical method from servers to clients. It is also used to cross-check clocks and control errors due to equipment or propagation failures.</p>
NWIP	<p>NetWare/IP. Allows total or partial replacement of the IPX transport subsystem with the industry-standard TCP/IP subsystem, in a NetWare network. The following constitute the core components of the technology:</p> <ul style="list-style-type: none"> ♦ Communication between the NetWare/IP server and the Domain SAP/RIP Service (DSS) for <ul style="list-style-type: none"> - Retrieval of configuration parameters - Registration of SAP and RIP information - SAP/RIP database synchronization ♦ Synchronization of the NetWare/IP server with the DSS database with respect to SAP/RIP information ♦ Communication between secondary DSS and primary DSS to synchronize the SAP/RIP database on the two servers
OSPF	<p>Open Shortest Path First. A protocol in the TCP/IP protocol suite is an interior gateway protocol algorithm and is proposed as a standard for the Internet. OSPF incorporates least-cost routing, multipath routing, load balancing, and efficient bandwidth utilization.</p>
POP3	<p>Post Office Protocol 3. A protocol used for interacting with a central mailbox server. It is a client/server protocol used to receive e-mail. The protocol holds the e-mail messages in the Internet server. Periodically, you can download the messages from the server.</p>
RARP	<p>Reverse Address Resolution Protocol. A protocol in the TCP/IP protocol suite that is used to determine a software address based on a hardware address. This protocol is often used by diskless workstations during startup.</p>

TCP/IP Protocol	Description
RIP	Routing Information Protocol. A protocol in the NetWare protocol suite that automates the process of updating routing tables. Routing is the process of moving network packets between separate networks. With RIP, when one router learns about changes in its routes, it broadcasts this information to neighboring routers so they can update their routing tables. As a result of RIP, if a network component fails (such as a router or a phone line), the other network components can inform each other of alternate routes. When the faulty component is repaired, the network changes back to the previous condition.
SSL	SSL is an open, nonproprietary protocol. It has been submitted to the W3 Consortium (W3C) working group on security for consideration as a standard security approach for World Wide Web browsers and servers on the Internet.
SLP	Service Location Protocol. This protocol provides a scalable framework for the discovery and selection of network services. Using this protocol, computers using the Internet no longer need as many static configurations of network services for network-based applications.
SMTP	Simple Mail Transfer Protocol. The application layer protocol that e-mail clients and servers use to exchange e-mail messages with each other.
SNMP	Simple Network Management Protocol. A protocol in the TCP/IP protocol suite that enables you to monitor a network from a single network management station called an SNMP Manager. From an SNMP Manager, you can make inquiries to another network device called the SNMP Agent. The SNMP Agent can be a TCP/IP host, router, terminal server, or another SNMP Manager. The information you can request from an SNMP Agent is contained in the MIB of that TCP/IP host. RFC 1066 (http://www.isi.edu/in-notes/rfc1066.txt) (Internet standard MIB) defines the types of objects that can be in an SNMP Agent MIB. These objects include network and hardware addresses, counters, and statistics, as well as routing and Address Resolution Protocol tables. Different vendors might not support all data types within their MIB or might include other information not defined within the RFC.
TCP	Transmission Control Protocol. This primary Internet transport protocol accepts messages of any length from an upper-layer protocol and provides full-duplex, acknowledged, connection-oriented, flow-controlled transport.
TELNET	Protocol in the TCP/IP suite that governs character-oriented terminal traffic.
TFTP	Trivial File Transfer Protocol. TCP/IP protocol commonly used for software downloads.
UDP	User Datagram Protocol. A protocol similar to TCP that provides connectionless, nonguaranteed transport services. UDP accepts and transports datagrams from an upper-layer protocol. Unburdened by the overhead of establishing and removing connections, controlling data flow, and performing other TCP functions, UDP usually provides a faster data conduit than TCP. For these reasons, and because it is easier to implement, UDP is the transport method of choice for many upper-layer protocols.

ZENWorks Management and Monitoring Services Database

ZENworks® for Servers (ZfS) provides a centralized Common Information Model (CIM)-compliant Sybase* database on the Management and Monitoring Services management server. The database serves as a repository for server and network data that can be displayed or formatted in various ways to provide you with exactly the information you need to manage your network.

The following sections provide information on understanding and using the ZENworks database:

- ◆ [“Understanding the ZfS Database” on page 289](#)
- ◆ [“Backing Up the Database” on page 290](#)
- ◆ [“Changing Database Passwords” on page 290](#)

Understanding the ZfS Database

The ZfS database consists of files located in the \ZENWORKS\MMS\DB directory on the management server. The ZfS data is stored in the following logical database:

- ◆ Topology/alarm database containing topology, alarms, and map information associated with the following files:
 - ◆ MW.DB
 - ◆ MW1.DB
 - ◆ MW2.DB
 - ◆ MW3.DB

The MW.LOG file in the \ZENWORKS\MMS\DB subdirectory saves your transaction information with the database files.

Running the Database

The database is run using the MGMTDBS.NCF file (located in the \SYSTEM directory on a server volume), which is executed from AUTOEXEC.NCF.

IMPORTANT: Ensure that the database is running as long as the ZfS services are running.

Database Caching

Increasing the database cache improves the database performance. The default database cache size is 48 MB. You can increase the cache size to an optimum level depending on the server memory. To increase the cache size, modify the **-c** option in SYS:\SYSTEM\MGMTDBS.NCF. For example, **-c 64M** sets the cache size to 64 MB. Reload the database after modifying the cache size.

Backing Up the Database

You should plan to regularly back up the ZfS database:

- ◆ [“Backing Up the Topology/Alarm Database” on page 290](#)

Backing Up the Topology/Alarm Database

From ConsoleOne, follow this procedure to back up the topology/alarm database:

- 1** Right-click the Site Server object > select Properties.
- 2** Select the Database Administration tab.
- 3** Enter the path of the directory to back up.

You can back up the database files to any volume on the management server only.

- 4** Click Apply.

ZfS sends a remote SQL command to store the file. The four MW*.DB and MW.LOG files are copied to the backup directory.

Changing Database Passwords

ZfS allows you to access the topology/alarm database at three different levels: Administrator account, Updater account, and Reader account. You can set passwords for any of the three different user accounts.

From ConsoleOne, follow this procedure to modify the database passwords:

- 1** Right-click the Site Server object > select Properties.
- 2** Select the Change Database Passwords tab.
- 3** Enter the new passwords and confirm.
- 4** Click Apply.

ZfS sends a remote SQL command to change the passwords of appropriate user objects in the database. The passwords are also stored in the Novell eDirectory

12

Using Reports in Management and Monitoring Services

The ZENworks[®] for Servers (ZfS) Management and Monitoring Services provide the following predefined reports:

- ◆ Topology Reports
- ◆ Alarm Reports
- ◆ Health Reports

The following sections describe the available reports and provide procedures for customizing and generating the reports:

- ◆ [“Understanding Management and Monitoring Services Reports” on page 291](#)
- ◆ [“Managing Reporting” on page 296](#)

Understanding Management and Monitoring Services Reports

The following sections describe each predefined report available in Management and Monitoring Services:

- ◆ [“About the Topology Reports” on page 291](#)
- ◆ [“About the Alarm Reports” on page 294](#)
- ◆ [“About the Health Reports” on page 295](#)

About the Topology Reports

The topology reports provide information about the topology of a selected ZfS site or segment. There are two types of topology reports you can generate: site-level reports and segment-level reports. The site-level reports provide details about the discovered devices on each segment in the ZfS site. The segment-level reports provide information about the discovered devices on the selected network segment.

Prior to generating the reports, you will need to perform a few operations. For more information see [“Prerequisites for Generating the Reports” on page 292](#).

There are five predefined topology reports:

- ◆ [“Computer Systems by Segment Report” on page 292](#)
- ◆ [“NCP Servers Report” on page 292](#)
- ◆ [“Router Report” on page 293](#)
- ◆ [“Segment Report” on page 293](#)

- ◆ “Segment Topology Report” on page 293

The NCP Servers report is available only at the site level.

Prerequisites for Generating the Reports

Because Crystal Reports is invoked by DLLs on the system, you need to install the Sybase ODBC driver. To check if the driver is installed:

- 1** From the desktop Start menu, click Settings > Control Panel > ODBC Data Source.
 - 1a** In the System Data Source Name (DSN) pane click Add.
 - 1b** Select the Adaptive Server Anywhere driver. You must install Adaptive Server Anywhere if you do not have it on your system. You can install it from the SYBASE.ZIP file at COMPANIONCD\ODBC\SYBASE*.*
- 2** If you have an older version of ZfS, you will need to uninstall it and install the latest version of ZfS before you can run the reports.

To uninstall the previous version:

- 2a** From the desktop Start menu, click Settings > Control Panel > Add/Remove Programs.
- 2b** Select ConsoleOne from the list and remove it.

If you have already installed the latest version, then delete the zenSnapins.jar file from CONSOLEONE\LIB\ZEN.

- 3** You will need at least MDAC 2.6 SP1 (Microsoft Data Access Component) for running Crystal Reports, particularly on a Windows NT machine. Check the version of MDAC on your box: select Control panel > ODBC Data sources > the About tab pane. The minimum version required is 3.520.7326.0. If the version you have does not match the minimum requirement, you need to upgrade the ODBC core components by downloading from [Microsoft site \(http://microsoft.com/data/download.htm\)](http://microsoft.com/data/download.htm).

Computer Systems by Segment Report

This report lists the number of computer systems on the selected segment. If the report is generated at the site level, the report lists the number of systems on each segment. For each segment, the report provides the following information about each connected computer system:

- ◆ Segment Name
- ◆ Segment Type
- ◆ Total nodes on a segment
- ◆ Node Name
- ◆ Node Address
- ◆ Services
- ◆ MIBs

NCP Servers Report

This report lists the following information for each server on the selected ZfS site:

- ◆ Server Name
- ◆ Total NCP servers on the site

- ◆ Server Label
- ◆ Server Address
- ◆ Labels (other names by which the server is known)
- ◆ MIBs

Router Report

This report provides the following information for each router on the selected ZfS segment or site:

- ◆ Total number of routers on the segment or site
- ◆ IPX Address
- ◆ Bound Segments
- ◆ Services
- ◆ MIBs
- ◆ IP Address
- ◆ MAC Address

Segment Report

This report lists the number of computer systems on the selected segment (segment level) or on all segments in the ZfS site (site level). For each segment, the report provides the following information about the systems connected to the segment:

- ◆ Segment Name
- ◆ Segment Type
- ◆ Total segments on the site
- ◆ IP configuration
- ◆ IPX configuration
- ◆ Total nodes on the segment

Segment Topology Report

This report provides information about the routers and bridges on a selected ZfS segment or site.

For each router, the report provides the following information:

- ◆ Router Name
- ◆ IP Address
- ◆ IPX Address
- ◆ MAC Address
- ◆ Bound Segment

For each bridge, the report provides the following information:

- ◆ Bridge Name
- ◆ Bridge Type
- ◆ Number of Ports
- ◆ Port: MAC Address and Bound Segment

About the Alarm Reports

The alarm reports provide information about the alarms received by the ZfS server. There are two types of alarm reports you can generate: Alarm details report and Alarm summary report.

Prerequisites for Generating the Reports

Because Crystal Reports is invoked by DLLs on the system, you need to install the Sybase ODBC driver. To check if the driver is installed:

- 1** From the desktop Start menu, click Settings > Control Panel > ODBC Data Source.
 - 1a** In the System Data Source Name (DSN) pane click Add.
 - 1b** Select the Adaptive Server Anywhere driver. You must install Adaptive Server Anywhere if you do not have it on your system. You can install it from the SYBASE.ZIP file at `COMPANIONCD\ODBC\SYBASE*.*`
- 2** If you have an older version of ZfS, you will need to uninstall it and install the latest version of ZfS before you can run the reports.

To uninstall the previous version:

- 2a** From the desktop Start menu, click Settings > Control Panel > Add/Remove Programs.
- 2b** Select ConsoleOne from the list and remove it.

If you have already installed the latest version, then delete the zenSnapins.jar file from `CONSOLEONE\LIB\ZEN`.

- 3** You will need at least MDAC 2.6 SP1 (Microsoft Data Access Component) for running Crystal Reports, particularly on a Windows NT machine. Check the version of MDAC on your box: select Control panel > ODBC Data sources > the About tab. The minimum version required is 3.520.7326.0. If the version you have does not match the minimum requirement, you need to upgrade the ODBC core components by downloading from the [Microsoft site \(http://microsoft.com/data/download.htm\)](http://microsoft.com/data/download.htm).

Alarms Details Report

This report lists Information of the alarms on the site. The report is generated based on the customized settings. The report provides the following information about each connected computer system:

- ◆ Alarm Severity
- ◆ Affected object name
- ◆ Source address
- ◆ Alarm state
- ◆ Alarm category

- ◆ Alarm generator
- ◆ Alarm time
- ◆ Alarm owner
- ◆ Alarm type
- ◆ Alarm summary

Alarms Summary Report

This report generates a brief summary of the alarms on the site. It provides a graphical representation of the distribution of alarms, for the selected number of days. The report provides the following information about each connected computer system:

- ◆ Alarm Severity
- ◆ Alarm Category
- ◆ Alarm Owner
- ◆ Alarm state
- ◆ Top alarm types
- ◆ Top affected objects
- ◆ Top source address

About the Health Reports

The Health Reports provide information about the overall health of a specified ZfS site or network segment. Each health report is based on a predefined health profile. The health profiles define the trend parameters that are used to calculate the overall health of the segment or site. There are five predefined health profiles:

- ◆ [“NetWare Server Profile” on page 295](#)
- ◆ [“Microsoft Windows Profile” on page 295](#)
- ◆ [“Ethernet Network Profile” on page 296](#)
- ◆ [“Token Ring Network Profile” on page 296](#)
- ◆ [“FDDI Network Profile” on page 296](#)

In addition, you can modify any of the existing profiles or create your own health report profiles. See [“Customizing a Health Profile” on page 297](#) or [“Adding a New Health Profile” on page 298](#).

NetWare Server Profile

Reports generated using this profile provide graphs of the following trend parameters and use these parameters to calculate the overall health of the NetWare servers in the selected atlas, segment, or page:

- ◆ Cache Buffers
- ◆ Cache Hits
- ◆ CPU Utilization
- ◆ Volume Free Space

Microsoft Windows Profile

Reports generated using this profile use the following trend parameters to calculate health:

- ◆ Cache Hits
- ◆ CPU Utilization
- ◆ Disk Free Space
- ◆ Available Memory

In addition, reports generated using this profile contain trend graphs for the following parameter:

- ◆ Logged in Users

Ethernet Network Profile

Reports generated using this profile use the following trend parameters to calculate overall health:

- ◆ Total Errors
- ◆ Network Utilization

In addition, reports generated using this profile contain trend graphs for the following parameters:

- ◆ CRC error packets
- ◆ Undersized packets
- ◆ Oversized packets
- ◆ Fragmented packets
- ◆ Jabbers

Token Ring Network Profile

Reports generated using this profile use the following trend parameters to calculate overall health. In addition, reports generated using this profile contain also contain trend graphs for the following parameters:

- ◆ Network Utilization
- ◆ Total Errors

FDDI Network Profile

Reports generated using this profile use the following trend parameters to calculate overall health:

- ◆ Total Errors
- ◆ Network Utilization

In addition, reports generated using this profile contain trend graphs for the following parameters:

- ◆ CRC error packets
- ◆ Undersized packets
- ◆ Oversized packets
- ◆ Lost frame errors

Managing Reporting

The following sections provide procedures for customizing, generating, printing, and exporting the ZfS reports:

- ◆ [“Managing the Topology Reports” on page 297](#)
- ◆ [“Managing the Server Management Health Reports” on page 297](#)

Managing the Topology Reports

You can generate two types of topology reports: site-level reports and segment-level reports. The site-level reports provide details about the discovered devices on each segment in the ZfS site. The segment-level reports provide information about the managed devices on the selected network segment. Note that the NCP Servers report is available only at the site level.

The following section describes how to generate, print, and export a topology report.

Generating a Topology Report

To generate a topology report:

- 1** Select the ZfS site object (to generate a site-level report) or a network segment object (to generate a segment-level report) > click Reports.
- 2** Select the report you want to generate > click Run Selected Report.
- 3** To print the report, click File > Print.

or

To export the report, click File > Export.

Managing the Server Management Health Reports

The server management component provides five standard profiles that you can use to generate health reports. You can set up reports based on these standard profiles or you can customize these profiles or create your own profiles on which to base your reports. For information about the standard health profiles, see [“About the Health Reports” on page 295](#).

This section contains the following tasks:

- ◆ [“Customizing a Health Profile” on page 297](#)
- ◆ [“Adding a New Health Profile” on page 298](#)
- ◆ [“Creating and Scheduling Health Reports” on page 298](#)
- ◆ [“Viewing and Printing a Health Report” on page 299](#)
- ◆ [“Running a Health Report” on page 300](#)
- ◆ [“Calculating the Overall Health” on page 300](#)

Customizing a Health Profile

To customize a health profile:

- 1** Right-click the ZfS site object > click Properties.
- 2** Select the Health Profiles tab.

- 3 Select the health profile you want to customize > click Edit.

The Edit Profile dialog box is displayed. This dialog box contains a list of the parameters that can be used to calculate the overall health of the device or segment to which the profile is applied.

- 4 Specify the directory location to which reports generated using this profile should be published by entering a value in the Publish Directory field.

To browse for a directory, click the Browse button (...).

- 5 Modify the parameters that are used to calculate health by checking or unchecking the In Health Calculation check box next to each parameter. For more information on the parameters that are used in health calculation see, [“About the Health Reports” on page 295](#).

- 6 Rank the importance of each trend parameter in calculating health by entering a number in the Weight field for each parameter you checked to include in the health calculation.

You can enter any whole number in the Weight field. The system will use the weights to determine how important the parameter is in calculating overall health. The larger the number, the more weight the parameter is given in calculating health.

- 7 Modify which parameters to render graphically in the health report by checking or unchecking the Show Trend on Report check box next to each parameter.

- 8 To save your changes, click OK.

Adding a New Health Profile

To add a new health profile:

- 1 Right-click the ZfS site object > click Properties.

- 2 Select the Health Profiles tab.

- 3 Click New.

The New Profile dialog box is displayed.

- 4 Enter a name for the new profile in the Name field.

- 5 Select the type of device or segment to which the profile applies from the Type drop-down list > click OK.

The Edit Profile dialog box is displayed.

- 6 Specify the directory location to which reports generated using this profile should be published by entering a value in the Publish Directory field.

To browse for a directory, click the Browse button (...).

- 7 Select the parameters you want to use to calculate health for reports generated using this profile by clicking the In Health Calculation check box next to the appropriate parameters. For more information on the parameters that are used in health calculation see, [“About the Health Reports” on page 295](#).

- 8 For each parameter you selected to include in the health calculation, indicate how important the parameter is in calculating overall health by entering a value in the Weight column.

You can enter any whole number in the Weight field. The system will use the weights to determine how important the parameter is in calculating overall health. The larger the number, the more weight the parameter is given in calculating health.

- 9 For each parameter that you want to be represented graphically in associated health reports, click the Show Trend on Report check box.
- 10 Click OK.

Creating and Scheduling Health Reports

To create and schedule a health report:

- 1 Right-click the container object > click Properties.
- 2 Select the Health Reports tab.
- 3 Click New.
The Edit Report dialog box is displayed.
- 4 Enter a name for the report in the Name field.
- 5 Select the profile to use when generating the report by selecting a value from the Profile drop-down list.
- 6 Indicate how often you want to generate the reports by selecting a value from the Period drop-down list.
You can choose to generate reports daily, weekly, or monthly.
- 7 Set the time and date you want the reports generated by selecting or entering the appropriate values in the Start Time, Day of the Week, and/or Day of the Month fields.
The available fields will depend on the period you selected.
- 8 Click OK.

The report will be generated at the date and time you entered and stored in the directory specified in the associated report profile. For information on viewing the reports, see [“Viewing and Printing a Health Report” on page 299](#).

Editing, Scheduling, and Deleting Health Reports

To edit and schedule a health report:

- 1 Right-click the atlas, page, or segment > click Properties.
- 2 Select the Health Reports tab.
- 3 Click Edit.
The Edit Report dialog box is displayed. Edit the required information
- 4 Click OK.

IMPORTANT: If you want to edit the schedule time of the report, it is recommended that you create a new report with the changed schedule time or delete the report.

To delete a health report:

- 1 Right-click the atlas, segment, or page > click Properties.
- 2 Select the Health Reports tab.
- 3 Click Delete.
- 4 Click OK.

Viewing and Printing a Health Report

After you create a health report, the report will be automatically generated on the day and time you specified. You can view the reports using a Web browser to open the INDEX.HTM file in the directory that is designated as the publish directory in the associated report profile.

IMPORTANT: Before you can view the health reports you must install Java* plug-in 1.3.1_01. You can get this plug-in from Sun Microsystems, Inc.

To view a health report:

1 Browse to the directory where the health reports for the associated profile are stored.

2 Use your browser to open the INDEX.HTM file.

The INDEX.HTM file is a Java file containing all reports that are stored in the directory. The left column of the INDEX.HTM file lists report hierarchy.

3 Click the plus sign next to the profile that is associated with the reports you want to view.

The profile object expands to display a list of container objects.

4 Click the plus sign next to the container object associated with the reports you want to view.

The object expands to display a list of report names associated with the object.

5 Click the plus sign next to the report you want to view.

The object expands to display a list of individual report instances. For example, a report that is scheduled to run daily will have a report instance for each day. The reports are named by date and time. For example, 2000.09.09_11.15.10_PDT is the name assigned to a report generated on September 9, 2000 at 11:15:10 Pacific daylight time.

6 Click the plus sign next to the report name to display a list of individual report pages.

The number of individual report pages depends on what report profile you selected and the object where you generated the report. For example, if you generated a report at the segment level using the Ethernet Network profile, there will only be one report page for the segment. If you generated a report at the site level using the Ethernet Network profile, there will be a report page for each Ethernet segment within the site. If you generated a report at the segment level using the NetWare Server profile, there will be a separate report page for each NetWare server on the segment.

7 Click an individual report page to display the health report in the right frame.

The top of the report displays statistical information about the segment or server and provides a calculation of overall health. The parameters used to determine overall health are defined in the associated health report profile. The bottom of the profile displays trend graphs depicting the overall performance of the server or segment. See [“About the Health Reports” on page 295](#) for a list of the parameters tracked and graphed in each of the standard profiles.

8 To print the report, click the Print Report button at the bottom of the left frame.

Running a Health Report

Although Health Reports are usually scheduled to run at a specified time of the day, week, or month, you may occasionally want to generate a Health Report on demand. To generate a Health Report on demand:

1 Right-click the atlas, segment, or page > click Properties.

2 Select the Health Reports tab.

3 Select the report you want to generate > click Now.

The report is saved to the directory specified in the report profile. See [“Viewing and Printing a Health Report” on page 299](#).

Calculating the Overall Health

Overall health is calculated using the following parameters:

- ◆ Attributes selected for health calculation.
- ◆ Associated weights assigned to each attribute.

You can only associate weights, which are used for health calculations.

- ◆ Values for each attribute

Yellow threshold (YT), Red threshold (RT), and maximum value (maxValue).

- ◆ Global threshold values

Global Green threshold (GG) is 100, Global Yellow threshold (GY) is 66, and Global Red threshold (GR) is 33.

Health Calculation

For each of the attribute used in overall health calculation, sample values based on the schedule specified while generating the reports are collected. These sample values are normalized using global thresholds and attribute thresholds, where Global Green is 100, Global Yellow is 66, and Global Red is 33. The global Green range = global Green - global Yellow; the global Yellow range = global Yellow - global Red; and the global Red range = global Red.

Normalization Formula

Normalized Value = Global Threshold - ((value - attribute Threshold)/(attribute Threshold Range) * (Global Range))

if (value > attribute's RED threshold)

global Threshold = global Red

attribute threshold = attribute Red threshold

attribute threshold range = attribute max Value - attribute Red threshold

global Range = global Red range

if (value > attribute's Yellow threshold)

global threshold = global Yellow

attribute threshold = attribute Yellow threshold

attribute threshold range = attribute Red - attribute Yellow

global range = global Yellow range

if (value > 0)

global threshold - ((value)) / (attribute threshold range) * (global range)

global threshold = global Green

attribute threshold Range = attribute Yellow threshold

global range = global Green range

Each of these may have an associated weight attached to it, which is configured in the respective profiles. Each of these attribute samples is then multiplied by the corresponding weights using the formula:

$$\text{value} = \text{value} * \text{attributeWeight} / \text{TotalWeight};$$

where — value is the particular sample after normalization, attributeWeight is the weight associated with the attribute and the TotalWeight is the total weight of all the attributes used in health calculation.

The other values displayed in Health Reports are based on the following calculations:

- ◆ Minimum Value = minimum of all the values in a given sample
- ◆ Maximum Value = Maximum value of all the values in a given sample
- ◆ Average Value = Sum of all the Values / no of Samples
- ◆ Trend is calculated based on the Slope:

$$\text{Slope} = (n * \sum x * y - \sum x * \sum y) / (n * \sum x * x - \sum x * \sum x)$$

where:

n = number of samples

x = time at which these samples were captured

y = trend values

if Slope > 0, then the trend is increasing

if Slope < 0, then the trend is decreasing

if Slope = 0, then the trend is steady

- ◆ Intercept = $(\sum y - \text{Slope} * \sum x) / n$
- ◆ Next Week Projection or Next Month Projection Value = Slope * time + Intercept
where time = Report Schedule Time (time when the report was scheduled) + 7 * 24 * 60 * 60 * 1000 for weekly Projection
- ◆ Report Schedule Time (time when the report was scheduled) + 30 * 24 * 60 * 60 * 1000 for Monthly Projection

WARNING: Exporting data in CSV (Comma Separated Value), Character Separated Value, and Tab SeparatedValue(TSV), does not export the complete data. As a workaround the you need to first export data in MS Excel format and then save it in the desired format.

If you export the generated reports in formats other than HTML or DHTML, the correct page numbers are not displayed. The page number is displayed incorrectly as Page -1 of 1, for all pages.

13

Using SNMP Community Strings

This chapter is referenced from the other sections. This section provides you information on SNMP, the SNMP community strings and how to configure SNMP community strings.

This section contains the following information:

- ♦ “About SNMP Community Strings” on page 303
- ♦ “Setting the SNMP Community Strings” on page 304

About SNMP Community Strings

SNMP is a protocol that offers network management services within the Internet suite of protocols.

SNMP uses a lightweight security mechanism whereby each protocol data unit (PDU) contains a community string. The SET community string is used in an SNMP Control operation and the GET community string is used in an SNMP Monitor operation.

SNMP community strings provide only a rudimentary form of security because they are transmitted in clear text in each SNMP request. Therefore, the community strings are exposed to any stations capable of monitoring an IP or Internetwork Packet Exchange™ (IPX™) network

Because Management Agent for NetWare and Management Agent for Windows are based on SNMP, all actions that are directed from network ConsoleOne to a server involve SNMP SET and GET requests from the manager to the agent. ConsoleOne® requests data from a managed server by issuing an SNMP GET request. An SNMP SET command is required to set server alarm thresholds or configuration parameters. In most cases, you are unaware of the underlying SNMP commands required to carry out requests you make from ConsoleOne, unless you are issuing requests on an SNMP-enabled device through the MIB Browser.

SNMP Security

Conducting management operations from ConsoleOne raises the issue of ensuring security. In particular, if unauthorized users configuration parameters on a server, severe performance problems or even sabotage network operations are encountered.

For these reasons, you should establish a scheme for changing the default community string PUBLIC to a proprietary community string used for communication between the management system and your SNMP agents.

Use the community keyword to define the community string to be used in the generated traps. The length of the community string is restricted to 32 bytes and cannot contain a space (except between quotes), tab, square bracket, equals sign, colon, semicolon, or number sign (#) characters. You can use Unicode* or International characters for the community string.

The default community string for Monitor operations is PUBLIC and for Control operations is null.

Setting the SNMP Community Strings

This section provides the following information:

- ◆ “Setting the SNMP Community String: NetWare Server” on page 304
- ◆ “Setting the SNMP Community String: ConsoleOne” on page 306
- ◆ “Setting Community Strings for an Individual Node” on page 306
- ◆ “Setting the SNMP Community String: Windows NT” on page 307

Setting the SNMP Community String: NetWare Server

You configure security access for SNMP communications using either SNMP LOAD command line parameters (NetWare 3.x/4.x/5.x/6 servers) or through INETCFG (NetWare 4.x/5.x/6 servers, or servers with NetWare MultiProtocol Router™ software installed).

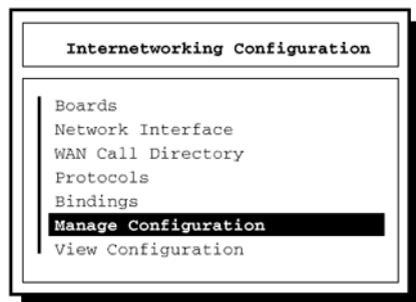
The following sections contain additional information to help you configure your NetWare servers:

- ◆ “Configuring Community String Options Using INETCFG” on page 304
- ◆ “Configuring Community String Options Using SNMP LOAD Commands” on page 305

Configuring Community String Options Using INETCFG

To configure the community string options using INETCFG:

- 1 At the server prompt, enter **LOAD INETCFG**.



- 2 From the Internetworking Configuration menu, click Manage Configuration > Configure SNMP Parameters > Monitor State.
- 3 Select one of the following options:

These options let you indicate how SNMP handles SNMP read operations coming from outside this server.

Option	Description
Any Community May Read	Allows all GET (read) commands no matter what community string is provided in the incoming read request.
Leave as Default Setting	Avoids changing the Monitor community string from its default (which is usually PUBLIC). The default Monitor Community can still be changed manually through SNMP command line options, as described in “Configuring Community String Options Using SNMP LOAD Commands” on page 305 .

Option	Description
No Community May Read	Allows GET (read) commands only for requests that are made by ConsoleOne that have logged in to the server with SUPERVISOR or OPERATOR privileges. Any community string provided in an incoming read request is ignored.
Specified Community May Read	Allows only GET (read) commands for requests that contain the name specified in the Monitor Community field. If you selected this option, type a name in the Monitor Community field, then press Enter. Enter the name of the community that is allowed to read management information. SNMP management stations that belong to this community can read the network management database.

4 Press Enter.

To change the Control community options, repeat Step 1 to Step 4 and choose the appropriate options for the community strings.

5 When you are finished, press Esc. If prompted, click Yes to save changes to the SNMP parameters > press Enter.

6 To return to the Internetworking Configuration menu, press Esc.

7 To exit INETCFG, press Esc.

8 Re-initialize the system.

To re-initialize, at the server prompt, enter **reinitialize system**.

Configuring Community String Options Using SNMP LOAD Commands

The LOAD command accepts the following SNMP option parameters:

- ◆ **MonitorCommunity:** Sets the community string for read-only (GET) access. The default value is PUBLIC. The syntax is as follows:

```
LOAD SNMP MonitorCommunity=community_name
```

- ◆ **ControlCommunity:** Sets the community string for read and write (GET and SET) access. By default, this community string is disabled.

The syntax is as follows:

```
LOAD SNMP ControlCommunity=community_name
```

These options set the community string for the indicated community. The following table shows examples of available settings:

IMPORTANT: Community strings are case sensitive.

Access Available to Requester	Read Only	Read/Write
Community name: "secret"	Load SNMP MonitorCommunity= <i>secret</i> or LOAD SNMP ControlCommunity= <i>secret</i>	LOAD SNMP ControlCommunity= <i>secret</i>
Community name: "str1" or "str2"	Load SNMP MonitorCommunity= <i>str1</i> and LOAD SNMP ControlCommunity= <i>str2</i>	
Any community name	Load SNMP MonitorCommunity="" or LOAD SNMP ControlCommunity=""	LOAD SNMP ControlCommunity=""

Setting the SNMP Community String: ConsoleOne

You set global community and trap target information using the SNMP property page associated with the site-level object. You can also customize the setting for a specific device using the SNMP property page of the device itself.

Setting Community Strings for an Individual Node

This section describes the procedure to set up the community strings for SNMP SET and GET operations on an individual node.

Typically, community strings are configured to be identical over all nodes in a network, or at least over a portion of the network. The default value for both SET and GET is public. The community strings are case sensitive.

By default, ZfS uses the public community string for SNMP GET and SET operations. You can configure a community string other than public on a node-by-node basis, or you can configure a community string globally on all SNMP-managed nodes. The community string that ZfS uses must match the string expected by the SNMP agent in the managed node; otherwise, the operation will fail.

To set up the community strings for SET and GET operations for an individual node:

- 1** From ConsoleOne, click the target SNMP-manageable node.
- 2** Right click the node > SNMP Settings.
- 3** Type the community string.

ZfS uses this community string for SET and GET operations when communicating with the device.

- 4** Click OK.

Setting the SNMP Community String: Windows NT

You configure security access for SNMP communications on Windows NT servers using the Network applet in the Windows NT Control Panel. For detailed information, refer to your Windows NT documentation or online help.

You must load the Microsoft* SNMP Service on your Windows NT servers. The SNMP community string setting must be the same as the SNMP community string setting on your ConsoleOne.

A

Documentation Updates

This section contains information on documentation content changes that have been made in the *Administration* guide for Management and Monitoring Services since the initial release of Novell® ZENworks® for Servers (ZfS) 3. This information will help you to keep current on updates to the documentation.

If you have purchased ZfS 3.0.2 and have not used or installed ZfS 3 or ZfS 3 SP1, you do not need to review this section.

All changes that are noted in this section were also made in the documentation. The documentation is provided on the Web in two formats: HTML and PDF. The HTML and PDF documentation are both kept up-to-date with the documentation changes listed in this section.

The documentation update information is grouped according to the date the documentation updates were published. Within a dated section, the changes are alphabetically listed by the names of the main table of contents sections for Management and Monitoring Services.

If you need to know whether a copy of the PDF documentation you are using is the most recent, the PDF document contains the date it was published on the front title page or in the Legal Notices section immediately following the title page.

The documentation was updated on the following dates:

- ♦ “November 8, 2004” on page 309
- ♦ “June 27, 2003” on page 309
- ♦ “April 15, 2003” on page 310
- ♦ “December 19, 2002” on page 311
- ♦ “September 27, 2002” on page 311
- ♦ “May 17, 2002” on page 312

November 8, 2004

Throughout the guide, graphics have been updated with font changes and newer icons. No notable content changes were made to any graphics.

June 27, 2003

Changes were made to the following sections:

- ♦ [Understanding Alarm Management](#)
- ♦ [Using Reports in Management and Monitoring Services](#)

Understanding Alarm Management

The following changes were made in this section:

Location	Change
“Troubleshooting the Alarm Management System” on page 123	<p>Flag values in the following paragraph have been corrected from True and False to be Yes and No. Also, “The default value of the flag is set to Yes” sentence was added to the paragraph.</p> <p>When AMS receives an unsolicited SNMP trap from an agent, it locates the appropriate alarm template for the trap-type object that is defined in the MIB of the device. If the alarm template is not available, the AMS checks the IgnoreUnknownTrap flag in the <install_volume>\<install_dir>\ZENWorks\MMS\MWServer\Properties\Alarmmanager.properties file. If the flag value is set to Yes the alarm is ignored. If the flag value is set to No the alarm is archived in the database as an unknown trap. The default value of the flag is set to Yes.</p>

Using Reports in Management and Monitoring Services

The following changes were made in this section:

Location	Change
“Creating and Scheduling Health Reports” on page 298	<p>Rephrased Step 1 from:</p> <ol style="list-style-type: none"> 1. Right-click the ZfS site object or a container object > click Properties <p>to:</p> <ol style="list-style-type: none"> 1. Right-click the container object > click Properties. <p>because you cannot create or schedule a health report from the site object.</p>

April 15, 2003

Changes were made to the following sections:

- ◆ [Understanding Network Discovery and Atlas Management](#)
- ◆ [Understanding Server Management](#)

Understanding Network Discovery and Atlas Management

The following changes were made in this section:

Location	Change
“Using Command Line Options for IPGROPER” on page 85	Added this new section.
“Displaying Server Configuration Information” on page 137	Included information about performing probe manageability to view the three categories of server data.

Understanding Server Management

The following changes were made in this section:

Location	Change
“Using Command Line Options for IPGROPER” on page 85	Added this new section.
“Displaying Server Configuration Information” on page 137	Included information about performing probe manageability to view the three categories of server data.

December 19, 2002

Changes were made to the following sections:

- ◆ [Understanding Network Discovery and Atlas Management](#)

Understanding Network Discovery and Atlas Management

The following changes were made in this section:

Location	Change
“Configuring the Java Processes” on page 97	Added this new section on configuring Java processes in setting up Discovery.

September 27, 2002

Changes were made to the following sections:

- ◆ [Understanding Network Discovery and Atlas Management](#)
- ◆ [Understanding Alarm Management](#)

Understanding Network Discovery and Atlas Management

The following changes were made in this section:

Location	Change
“Setting Up Discovery” on page 88	Added the following information about configuring the Consolidator to discover multiple IP addresses and a single MAC address connected to a more than one segment.

Understanding Alarm Management

The following changes were made in this section:

Location	Change
“Sending SMTP Mail Notification” on page 118	<p>Following changes were made in this section:</p> <ul style="list-style-type: none"> ◆ Changed the description of the <code>n</code> parameter. ◆ Added the following variable parameters: <code>p</code>, <code>h</code>, and <code>nnnX</code>.
“Launching an External Program” on page 119	<p>Following changes were made in this section:</p> <ul style="list-style-type: none"> ◆ Changed the description of the <code>n</code> parameter. ◆ Added the following variable parameters: <code>p</code>, <code>h</code>, and <code>nnnX</code>.

May 17, 2002

Changes were made to the following sections:

- ◆ [Using SNMP Community Strings](#)
- ◆ [Understanding Server Management](#)
- ◆ [Using the MIB Tools](#)
- ◆ [Understanding Traffic Analysis](#)
- ◆ [Protocol Decodes Suites Supported by ZfS](#)

Using SNMP Community Strings

The following changes were made in this section:

Location	Change
Chapter 13, “Using SNMP Community Strings,” on page 303	This chapter was added. This chapter provides information about SNMP community strings, and how to configure the SNMP community strings on servers.

Understanding Server Management

The following changes were made in this section:

Location	Change
Chapter 5, "Understanding Server Management," on page 125	Information on community strings, securing SNMP transaction, and defining community string for NetWare Management Agent have been removed and re-written in Chapter 13, "Using SNMP Community Strings," on page 303.

Using the MIB Tools

The following changes were made in this section:

Location	Change
Chapter 6, "Using the MIB Tools," on page 165	Information on setting community strings on an individual node, and the SNMP related information for working with MIB tools have been removed and re-written in Chapter 13, "Using SNMP Community Strings," on page 303.
Chapter 6, "Using the MIB Tools," on page 165	Two keywords HELP and HELPTAG have been added for the trap annotations.
Chapter 6, "Using the MIB Tools," on page 165	Information on updating vendor MIBs and updating ZfS MIBs are removed.

Understanding Traffic Analysis

The following changes were made in this section:

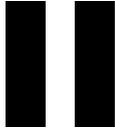
Location	Change
Chapter 8, "Understanding Traffic Analysis," on page 191	Information on configuring the SNMP parameters has been removed and re-written in Chapter 13, "Using SNMP Community Strings," on page 303.

Protocol Decodes Suites Supported by ZfS

The following changes were made in this section:

Location	Change
Chapter 10, "Protocol Decodes Suites Supported by ZfS," on page 281	The information for the NDS protocol under the NetWare Protocol Suite is changed.
Chapter 10, "Protocol Decodes Suites Supported by ZfS," on page 281	In the AppleTalk Protocol Suite, the following protocols are deleted: ARI, RPC, SLP, SMTP, SNMP, TCP, TELNET, TFTP, UDP.

Location	Change
Chapter 10, "Protocol Decodes Suites Supported by ZfS," on page 281	In the TCP/IP Protocol Suite, the following protocols are deleted: DIAG, NCP, RPC. These protocols are replaced with the following: IMAP, NFS, SSL.



Policy and Distribution Services

Novell® ZENworks® for Servers (ZfS) Policy and Distribution Services is a software, configuration, and behavioral management system for servers. Through Policy and Distribution Services, you can:

- ◆ Control the versions of software installed on servers throughout the network
- ◆ Define and enforce a standard configuration on any given set of servers
- ◆ Control the behavior of servers in given situations, such as downing a server, backing up volumes, managing thresholds exceeded, and so on

Policy and Distribution Services has three components:

- ◆ **Tiered Electronic Distribution (TED):** Simplifies data delivery and server policy implementation
- ◆ **Server Policies:** Simplifies configuration and management of your servers
- ◆ **Server Software Packages:** Simplifies the installation of software

These components are supported on servers for the following server platforms:

NetWare® 5.1 and NetWare 6
Windows* NT* 4.0 and Windows 2000
Linux* (Red Hat* 7.1, 7.2, 7.3, and 8)
Solaris* 8

You can administer Policy and Distribution Services using the following:

- ◆ **ConsoleOne® 1.3.5**, where you can create and configure ZfS objects and perform management tasks for Policy and Distribution Services.
- ◆ **ZfS Management Role in Novell iManager**, where you can perform management tasks for Policy and Distribution Services using iManager from any workstation where Internet Explorer 5.5 or later is available.

The Policy and Distribution Services documentation contains the following sections:

- ◆ **Chapter 14, “Configuring Policy and Distribution Services,” on page 317** (After installing ZfS 3.0.2 for the first time, use this section to complete a full configuration of your distribution system.)
- ◆ **Chapter 15, “Novell iManager,” on page 359**
- ◆ **Chapter 16, “Tiered Electronic Distribution,” on page 371**
- ◆ **Chapter 17, “Server Policies,” on page 455**
- ◆ **Chapter 18, “Server Software Packages,” on page 493**
- ◆ **Chapter 19, “Desktop Application Distribution,” on page 525**

- ◆ Chapter 20, “Security in Policy and Distribution Services,” on page 549
- ◆ Chapter 21, “Scheduling,” on page 565
- ◆ Chapter 22, “Variables,” on page 579
- ◆ Chapter 23, “ZENworks Database,” on page 587
- ◆ Chapter 24, “Reporting,” on page 599
- ◆ Appendix B, “Server Console Commands,” on page 617
- ◆ Appendix C, “Load/Unload Actions,” on page 623
- ◆ Appendix D, “Requirements for Server Software Packages,” on page 625
- ◆ Appendix E, “Registry Entries for Server Software Package Components,” on page 629
- ◆ Appendix F, “Using Server Software Packages to Delete Directories on Servers,” on page 633
- ◆ Appendix G, “Documentation Updates,” on page 637 (identifies where the documentation has been updated)

14

Configuring Policy and Distribution Services

To use Novell® ZENworks® for Servers (ZfS) Policy and Distribution Services effectively, you must correctly install and configure its components on your network. You should have already performed a basic installation of Policy and Distribution Services (see [“Installing Policy and Distribution Services on NetWare and Windows Servers”](#) under [“Installing ZENworks for Servers”](#) in the *Installation* guide).

The following sections provide you with the concepts, a [planning worksheet](#), and instructions to help you configure Policy and Distribution Services fully so that you can use its features to manage your network.

You can plan to configure all or just part of your distribution system, depending on the complexity of your network and how much you intend to learn by doing.

The information provided in the following sections will help you to add new Distributors as needed, finish installing the Subscriber software as needed, configure a Distributor’s routing hierarchy, create some Distributions, and send those Distributions. You can consult these sections at any time to add and configure new Distributors, or to add new Distributions.

- ◆ [“Planning Your Distribution System” on page 318](#)

In this planning section, you can use the [planning worksheet](#) to keep track of the decisions you will be making. Then you can easily perform your planned configurations from the information on the planning worksheet.

If you have a good understanding of Tiered Electronic Distribution (TED) in ZfS 3.0.2, you can skip this section.

- ◆ [“Configuring Your Distribution System” on page 337](#)

This section provides the steps for configuring Policy and Distribution Services.

If you skip the planning section, you can fill in the [planning worksheet](#) and continue with this section, or just continue with this section.

- ◆ [“Managing Your Distribution System” on page 349](#)

This section provides an overview on how you can manage your distribution system using ConsoleOne® and Novell iManager.

- ◆ [“Configuration Planning Worksheet” on page 349](#)

The planning worksheet contains basic information for each worksheet entry. It also contains links to where you can view more information to better understand a worksheet entry.

The worksheet should not be used in place of the procedures in [“Configuring Your Distribution System” on page 337](#), because the worksheet only contains information where planning is necessary. The worksheet does not contain information for procedures where planning is not required.

Planning Your Distribution System

Use these sections in the following order:

1. [“Overview of Policy and Distribution Services” on page 318](#)
2. [“Selecting Your Distributions” on page 320](#)
3. [“Understanding Your Network Topology” on page 324](#)
4. [“Are Additional Distributors Needed?” on page 325](#)
5. [“Other Subscribers To Be Installed?” on page 328](#)
6. [“Determining the Distribution Flow” on page 329](#)
7. [“Understanding Distribution Security” on page 331](#)
8. [“Determining the Channels for the Distributions” on page 333](#)
9. [“Determining Subscribers’ Subscriptions” on page 334](#)
10. [“Determining the Distribution Schedules” on page 335](#)

Overview of Policy and Distribution Services

Policy and Distribution Services contains three components:

- ◆ **Tiered Electronic Distribution** is a distribution system for your network.
 - ◆ It is a way to manage your network servers through the distribution of electronic data between servers.
 - ◆ It uses a tiered architecture for distribution efficiency. For example, workload sharing: one server can service many others, then each of those many servers can also service many more, and so on to any number of tiers.
 - ◆ It provides Distribution scheduling for efficient bandwidth usage, such as distributing during off-peak hours.
 - ◆ It provides security to prevent unauthorized tampering with the Distributions.
- ◆ **Server Policies** is a system for managing the configuration and behavior of your servers.
- ◆ **Server Software Packages** is a feature for automating the installation and upgrading of software on your servers.

TED is usually involved when using any of these components. Therefore, in planning how to configure Policy and Distribution Services, we will concentrate on understanding and configuring TED.

The following sections provide basic information that will help you to understand TED and what you will need to know to configure it:

- ◆ [“What Can You Distribute?” on page 318](#)
- ◆ [“How Is Data Distributed?” on page 319](#)
- ◆ [“What Will You Need To Know To Plan Your Distribution System?” on page 319](#)

What Can You Distribute?

The types of electronic data you can distribute using TED include:

Distribution Type	Explanation
File	Files and directories contained on the Distributor server's file system
FTP	Files and directories from an FTP source
HTTP	Content from an HTTP source
RPM	RPM packages for Solaris and Linux servers (but only for Solaris if RPM is installed to the Solaris machine)
Desktop Application	Desktop Application objects and files created in ZENworks for Desktops (ZfD)
Policy Package	Policies for controlling servers
Software Package	Server Software Packages for automatically installing or upgrading software on your servers

From this list, you can see that there is a variety of electronic data types that you can distribute to your servers. In later sections, you will be able to understand, create, and configure the Distributions for each type.

How Is Data Distributed?

TED sends Distribution files from Distributor servers to Subscriber servers. The basic distribution process is as follows:

1. Decide what you want to distribute.
2. Create the Distribution.
3. Create a Channel for the Distribution.
4. Determine which Subscriber servers need this Distribution.
5. Subscribe the Subscriber servers to the Distribution's Channel.
6. Make sure the applicable schedules are set (Build, Send, and Extract).
7. Send the Distribution by refreshing the Distributor, which causes the Distribution to be built according to the Distribution's Build schedule, and sent according to the Channel's Send schedule.
8. The Distribution is extracted on the Subscriber servers according to their Extract schedules.
9. The Distributions are used by the Subscriber servers according to the Distribution's type.

From this process, you can see that there are several components of TED that will need to be created and configured. For more information, see [“Understanding the Distribution Processes” on page 440](#) and [“The Basic Distribution Process” on page 372](#).

What Will You Need To Know To Plan Your Distribution System?

You will need to know the following in order to fully configure Policy and Distribution Services:

- ❑ The Distributions that you will want, including:
 - ◆ Whether you want to distribute server files, HTTP content, FTP content, or RPM packages

- ◆ If there are any ZfD desktop applications to be distributed (affects how you set up Subscriber objects when you have multiple trees)
 - ◆ Which policies you will need for managing your servers
 - ◆ What server software should have automated installation
- Whether you'll need additional Distributors
 - Whether you have both Novell eDirectory™ 8.x and NDS® 7.x in your environment, which adversely affects Distributors (a workaround is available)
 - How many databases you'll need for reporting purposes
 - Whether you need to complete installation of the Subscriber software to your servers
 - Which Subscribers will need which Distributions
 - Your network's topology (server platforms, slow WANs, firewalls, NATs, multiple trees, and so on)
 - Which types of Distribution security you'll need
 - The system resource and server behavior issues that TED might create
 - Whether you need to encrypt Distributions for certain servers
 - Whether you can use Subscriber Groups for channeling Distributions
 - How you want the Distributions to flow to the Subscriber servers (the tiered distribution model)
 - How you want to schedule the distribution processes to minimize network traffic, such as during business hours

To determine the above information, plan your configuration, and configure TED, continue with [“Selecting Your Distributions” on page 320](#).

Selecting Your Distributions

This section provides you with basic information for each type of Distribution.

You can build your distribution system incrementally by adding Distributions a few at a time, then adding Distributors as you need them.

There are seven Distribution types. Each has properties for determining how to build and extract a Distribution.

You can revisit this process at any time to add new Distributions.

Review the following Distribution type sections to select which ones you want to create at this time. [Planning worksheet](#) entries are provided for each Distribution type.

- ◆ [“File” on page 321](#)
- ◆ [“FTP” on page 321](#)
- ◆ [“HTTP” on page 321](#)
- ◆ [“RPM” on page 322](#)
- ◆ [“Software Package” on page 322](#)
- ◆ [“Desktop Application” on page 322](#)

- ◆ “Policy Package” on page 323

File

With this type you can select files and/or directories from the Distributor server’s file system for distribution, and select a destination location for extraction on the Subscriber.

A Distribution Wizard is available for automating the process of creating the File and FTP types of Distributions. For more information, see “Using the Distribution Wizard” under “Installing on NetWare and Windows Servers” in “Installing Policy and Distribution Services on NetWare and Windows Servers” in the *Installation* guide.

For information on the File type of Distribution, see “File” on page 398.

Determine whether you want to create a File Distribution at this time:

CONFIGURATION PLANNING WORKSHEET

Under **item 19**, enter the File type as a Distribution to be created. Also indicate the following:

- ◆ A name for the Distribution that indicates its purpose
 - ◆ Names of the servers that will need a File Distribution
-

FTP

With this type you can create a Distribution consisting of files from one or more FTP sources. Each source can contain one or more directories and/or files.

A Distribution Wizard is available for automating the process of creating the File and FTP types of Distributions. For more information, see “Using the Distribution Wizard” under “Installing on NetWare and Windows Servers” in “Installing Policy and Distribution Services on NetWare and Windows Servers” in the *Installation* guide.

For information on the FTP type of Distribution, see “FTP” on page 399.

Determine whether you want to create an FTP Distribution at this time:

CONFIGURATION PLANNING WORKSHEET

Under **item 19**, enter the FTP type as a Distribution to be created. Also indicate the following:

- ◆ A name for the Distribution that indicates its purpose
 - ◆ Names of the servers that will need an FTP Distribution
-

HTTP

With this type you can create a Distribution consisting of one or more HTTP sources. Each source can contain one or more target entries.

For information on the HTTP type of Distribution, see “HTTP” on page 400.

Determine whether you want to create an HTTP Distribution at this time:

CONFIGURATION PLANNING WORKSHEET

Under **item 19**, enter the HTTP type as a Distribution to be created. Also indicate the following:

- ◆ A name for the Distribution that indicates its purpose
 - ◆ Names of the servers that will need an HTTP Distribution
-

RPM

This is a UNIX platform Distribution. You can distribute Red Hat Package Manager (RPM) packages using the RPM Distribution.

For information on the RPM type of Distribution, see **“RPM” on page 400**.

Determine whether you want to create an RPM Distribution at this time:

CONFIGURATION PLANNING WORKSHEET

Under **item 19**, enter the RPM type as a Distribution to be created. Also indicate the following:

- ◆ A name for the Distribution that indicates its purpose
 - ◆ Names of the servers that will need an RPM Distribution
-

Software Package

A Server Software Package is created in ConsoleOne in the Server Software Package namespace. It is first created as a .spk file, then compiled into the .CPK file that is distributed.

For information on Server Software Packages, see **Chapter 18, “Server Software Packages,” on page 493**.

For information on the Software Package type of Distribution, see **“Software Package” on page 400**.

Determine the software packages you want to create at this time:

CONFIGURATION PLANNING WORKSHEET

Under **item 19**, enter the Software Package type as a Distribution to be created. Also indicate the following:

- ◆ A name for the Distribution that indicates its purpose
 - ◆ Names of servers that will need a Software Package Distribution
-

Desktop Application

Distributes ZENworks for Desktops (ZfD) Application objects and associated files to specified locations on the eDirectory tree and target Subscriber servers.

This Distribution type is not supported for Linux and Solaris servers.

For information on integration with ZfD, see **Chapter 19, “Desktop Application Distribution,” on page 525**.

For information on the Desktop Application type of Distribution, see “[Desktop Application](#)” on [page 401](#).

Determine whether you want to create a Desktop Application Distribution at this time:

CONFIGURATION PLANNING WORKSHEET

Under [item 3](#) and [item 20](#), indicate that you will have Desktop Application Distributions, and therefore each server that will be receiving Desktop Application Distributions must have its Subscriber object and NCP Server object on the same tree.

Under [item 19](#), enter the Desktop Application type as a Distribution to be created. Also indicate the following:

- ♦ A name for the Distribution that indicates its purpose
 - ♦ Names of the servers that will need a Desktop Application Distribution
-

Policy Package

Provides the mechanism for applying policies to servers. In previous versions of Policy and Distribution Services, policies were enforced through eDirectory object and container associations. With ZfS 3.0.2, policies are now distributed for enforcement on the receiving Subscriber servers.

Select from the following policies:

Policy	Description
Copy Files	Enables copying of files on a server from one location to another by using policy configurations.
NetWare Set Parameters	Specifies and optimizes selected NetWare® Set Parameters for a server or group of servers.
Scheduled Down	Schedules when a server should go down, and whether it should be brought back up automatically.
Scheduled Load/Unload	Automates the loading and unloading order of NLM™ and Java* Class processes for the selected servers, and for starting and stopping Windows services.
Search	Used in ZfS to enable the Distributor Agent to locate and use policies in the Service Location Package.
Server Down Process	Controls which processes to follow and which conditions to meet before downing a server.
Server Scripts	Automates script usage on your servers.
SMTP Host	Sets the TCP/IP address of the relay host that processes outbound Internet e-mail.
SNMP Community Strings	Allows you to receive and respond to SNMP requests.
SNMP Trap Targets	Sets SNMP trap targets for associated eDirectory objects for reporting purposes.
Text File Changes	Automates changes to text files.

Policy	Description
Tiered Electronic Distribution	Sets defaults for the Distributor and Subscriber objects.
ZENworks Database	Sets the DN for locating the ZENworks Database object. This policy must be in effect for Policy and Distribution Services to locate a database for logging successes and failures that are used in creating reports.
ZENworks for Servers	Contains basic configuration parameters for Policy and Distribution Services, such as status logging, defining the server console prompt for the Policy/Package Agent, setting its working path, and setting a database purging limit.

For more information on each policy, see [“Server Policy Descriptions” on page 461](#).

For information on policies and policy packages, see [Chapter 17, “Server Policies,” on page 455](#).

For more information on the Policy Package type of Distribution, see [“Policy Package” on page 401](#).

Determine whether you want to create a Policy Package Distribution at this time:

CONFIGURATION PLANNING WORKSHEET

Under [item 19](#), enter the Policy Package type as a Distribution to be created. Also indicate the following:

- ◆ Names of the policies
 - ◆ For each policy, names of servers that will need the policy
-

Understanding Your Network Topology

In order for you to efficiently manage Policy and Distribution Services, you will need to know your network’s topology. For example:

- ◆ What are your server platforms?
- ◆ How many servers do you have per platform?
- ◆ Where are your servers located in relation to WAN links and firewalls?
- ◆ Is Network Address Translation (NAT) being used?
- ◆ Where are your slow network links?

This type of information will be used to help you configure the best distribution management solution for your network.

Print a copy of the [“Configuration Planning Worksheet” on page 349](#). You will be instructed to fill in the worksheet when reviewing the remaining planning sections.

Obtain the following information concerning your network:

- Note the trees where you extended the schema for ZfS.

CONFIGURATION PLANNING WORKSHEET

Under **item 1**, enter the names of the trees in your network where you extended the schema for ZFS.

- Draw a diagram of your network structure.
You will use this diagram to determine distribution routes.
Indicate the following on your diagram:
 - ◆ Where slow links exist
 - ◆ The number of servers on each LAN (for Subscriber candidates)
 - ◆ The number of servers outside a firewall
 - ◆ The number of servers using NAT
- Draw tree diagrams that show how your trees are currently organized. Include the main containers, such as:
 - ◆ The containers that represent geographic locations (a physical tree design)
 - ◆ The containers that represent the corporate organization (a logical tree design)
 - ◆ The containers where servers reside (for Distributor and Subscriber candidates)
- Indicate the following on your tree diagrams:
 - ◆ Where servers are located that could be Distributors (NetWare, Windows, or UNIX servers that exceed the minimum ZFS requirements)
 - ◆ Containers where there are slow network connections
This should match where you indicated slow connections on your network diagram.
- Indicate the following on your network diagram:
 - ◆ Where the servers are located (as you just noted on the tree diagrams) that could be Distributors

Are Additional Distributors Needed?

When installing Policy and Distribution Services for the first time, you installed one Distributor with a database file. Generally, you'll need Distributors according to geographic locations or your corporate structure.

User your diagrams to determine whether you need to install additional Distributors.

IMPORTANT: Because Distributions belong exclusively to their Distributors, you will not be able to transfer its Distributions to another Distributor should you later change your mind about using your selected server as the Distributor. The Distributions would need to be re-created from scratch for another Distributor. For more information, see ["Deleting a Distributor Object and How Its Distributions Are Affected" on page 392](#).

After you've seen how your Distributor servers handle their Distribution building and sending workload, you can determine whether to add additional Distributors for spreading that workload.

CONFIGURATION PLANNING WORKSHEET

Under **item 2**, enter the names of the servers where you want to install the Distributor software.

You will also need to determine the following information for each Distributor:

- ◆ [“Determining Distributor Properties” on page 326](#)
- ◆ [“Determining ZfS Software Installation Paths” on page 326](#)
- ◆ [“Determining Whether a Distributor Server Will Host a ZENworks Database” on page 327](#)
- ◆ [“Configuring Distributors in a Mixed eDirectory Environment” on page 328](#)

Determining Distributor Properties

The following Distributor properties can be changed from the defaults during installation:

- ◆ **Object Name:** If you want to rename the Distributor object, we recommend that you maintain the server’s identity in the name, including the fact that it is a Distributor.
- ◆ **TED Container:** Plan on using the TED container where you previously installed TED objects.

If eDirectory is not installed on the Windows NT or Windows 2000 server that you want to be a Distributor, a default container object will not be displayed for that server during installation. Therefore, determine a TED container for that Distributor object.

- ◆ **Working Directory:** You can use a different volume, drive, or directory path for the Distributor’s working files than the default path.

Because the working directory has the potential to be quite large (depending on the size of the Distributions), make sure you have enough disk space.

The default volume on a NetWare server is SYS:. For NetWare servers we strongly recommend that you specify a different volume.

The default working directory path for NetWare and Windows servers is:

```
ZENWORKS\PDS\TED\DIST
```

For UNIX servers the path is:

```
usr/ZENworks/PDS/TED/Dist
```

The Distributor’s working directory is also used whenever a Distribution is created. A subdirectory is created under the working directory using the DN of the Distribution object.

For more information on the working directory, see [“Working Directories” on page 447](#).

CONFIGURATION PLANNING WORKSHEET

Under [item 7](#), enter property information for the Distributor that you want to be different than the defaults. This includes object names, containers for the object, and working directories.

Determining ZfS Software Installation Paths

ZfS uses the following default installation paths:

- ◆ **NetWare:** SYS:\ZENWORKS
- ◆ **Windows:** C:\ZENWORKS
- ◆ **Linux or Solaris:** usr/ZENworks

The Linux or Solaris path cannot be edited. However, you can use different paths for Distributors and Subscribers for NetWare and Windows servers.

IMPORTANT: During installation, ZfS updates .NCF files with installation path information. Because NetWare uses a DOS code page instead of a Windows code page, double-byte or extended characters cannot be used in the paths, or the .NCF files will not execute. Therefore, do not use double-byte or extended characters in any part of an installation path, including a NetWare volume name.

CONFIGURATION PLANNING WORKSHEET

Under **item 5**, enter the installation path information for the Distributor if it is different from the default path. Include the identities of the Distributors where you have different Distributor installation paths.

Under **item 6**, enter the installation path information for the Subscriber if it is different from the default path. Include the identities of the Subscribers where you have different Subscriber installation paths.

Determining Whether a Distributor Server Will Host a ZENworks Database

You can have multiple ZENworks databases in the tree, and you can install the database to both NetWare and Windows servers.

The database is used by Policy and Distribution Services to log successes and failures for the Server Policies or TED components. Policy and Distribution Services can function normally without a database, because it uses the ZFSLOG.DB file to only log information for reports. ZFSLOG.DB for Policy and Distribution Services does not contain any configuration information.

To determine whether you want each Distributor to have its own database, or have all Distributors share the same database, you need to determine how you want information reported. Consider the following to determine how many databases to have in the tree:

- ◆ **WAN Traffic:** TED does not perform a large number of database updates, so the actual impact on system resources should be minimal. The greatest impact could be the time it takes to perform the transaction. However, if you have slow WAN connections, you might not want database logging to occur over the WAN.
- ◆ **Multiple Distributors:** If you have multiple Distributors in the tree, you can have one database for each, or have them share one or more databases. The type of Distributor reporting you want should determine whether to have a separate database for each. For example, are your Distributors specialized in the types of Distributions they'll send?
- ◆ **Consolidated Reporting:** To have only one report for all of your TED information, install only one database object and file and have all TED Distributors log to that one file, regardless of WAN traffic considerations. Use the ZENworks Database policy (Service Location Package) to direct all Distributors to that database file.
- ◆ **Specialized Reporting:** You might want reports that are specific to a region or group of servers. You can install a database object and file for each region and have the Distributors in those regions or server groups log to that database. Use a separate ZENworks Database policy (Service Location Package) to direct each Distributor to its desired database file.

For more information, see [Chapter 23, "ZENworks Database," on page 587](#).

IMPORTANT: Make sure you select a server for the database where you are installing the Subscriber/Policies option. The Purge Database option in the ZENworks for Servers policy (Distributed Server Package) works only if the Policy/Package Agent software and the ZFSLOG.DB file are located on the same server.

CONFIGURATION PLANNING WORKSHEET

Enter the following information for each Database object to be created:

- ◆ Under **item 4**, enter the name of the Distributor server that will host the ZENworks Database files.
 - ◆ Under **item 9**, enter the installation path information that is different from the default path.
 - ◆ Under **item 10**, enter a name for the Database object, if different from the default.
 - ◆ Under **item 11**, enter the eDirectory container where the Database object should be created.
-

Configuring Distributors in a Mixed eDirectory Environment

In ZfS 3.0.2, Distributor servers must be able to authenticate to the eDirectory 8.x tree. If your network has both eDirectory 8.x and NDS 7.x installed, you must edit the TED.NCF file on each of your NetWare Distributor servers to ensure that they can authenticate to an eDirectory 8.x tree.

Select an IP address of any server in your tree that is using eDirectory 8.x. This can even be the IP address of the Distributor server itself, if the server is running eDirectory 8.x.

CONFIGURATION PLANNING WORKSHEET

Under **item 12**, enter the IP address of a server using eDirectory 8.x.

Other Subscribers To Be Installed?

When you first installed Policy and Distribution Services, you might not have installed the software to all of your servers. If you determined that you wanted to install the Subscriber software incrementally to your servers, you can complete another stage at this time.

In setting up a distribution system, not all of your Subscribers need to be installed and running. Subscriber servers can be added to the distribution system at any time.

The following Subscriber properties can be changed from the defaults during installation:

- ◆ **Object Name:** If you want to rename the Subscriber object, we recommend that you maintain the server's identity in the name, including the fact that it is a Subscriber.
- ◆ **TED Container:** Plan on using the TED container where you previously installed TED objects.

You can use the same context for all Subscriber servers.

If eDirectory is not installed on the Windows NT or Windows 2000 server that you want to be a Subscriber, a default container object will not be displayed for that server during installation. Therefore, determine a TED container for that Subscriber object.

- ◆ **Working Directory:** You can use a different volume, drive, or directory path for the Subscriber's working files than the default path.

Because the working directory has the potential to be quite large (depending on the size of the Distributions), make sure you have enough disk space. The default volume on a NetWare server is SYS:. For NetWare servers we strongly recommend that you specify a different volume.

You might need to provide different paths for your Subscriber servers. For example, SYS: for NetWare servers and D: for Windows servers. Variables can be used for path data, such as the volume/drive designation. For more information, see [Chapter 22, "Variables," on page 579](#).

The default working directory path for NetWare and Windows servers is:

```
ZENWORKS\PDS\TED\SUB
```

For UNIX servers the path is:

```
usr/ZENworks/PDS/TED/working/Sub
```

For more information on working directories, see [“Working Directories” on page 447](#).

CONFIGURATION PLANNING WORKSHEET

Under [item 3](#), enter the names of the servers where you want to install the Subscriber software at this time.

For each Subscriber to be installed, under [item 8](#), enter the property information that you want to be different than the defaults. This includes object names, containers for the object, and working directories.

Determining the Distribution Flow

The following sections provide information for determining distribution routes:

- ◆ [“Understanding Distribution Routes” on page 329](#)
- ◆ [“Selecting Subscribers for the Distribution Routes” on page 330](#)
- ◆ [“Configuring the Distribution Routes” on page 331](#)

For more detailed information, see [“Understanding Distribution Routing” on page 381](#).

Understanding Distribution Routes

Each Distributor has a routing hierarchy that provides it with a hierarchical path for sending its Distributions. The routing hierarchy contains a list of Subscribers. The hierarchy of Subscribers can be many levels deep.

Subscribers in a Distributor’s routing hierarchy do not need to also be recipients of the Distributions from that Distributor. A Subscriber can merely act as a proxy for the Distributor to pass Distributions to other Subscribers.

Not all Subscribers are needed in a routing hierarchy; only those that will be used to pass Distributions on to other Subscriber servers. Most of your network’s Subscriber servers will likely be end-node Subscribers; meaning, Subscribers that only receive and extract the Distributions.

The Distributor determines the most efficient route to any given Subscriber as follows:

1. The Distributor identifies the Subscriber that is to receive the Distribution.
2. The Distributor determines whether that Subscriber has a parent Subscriber.
3. If the Subscriber has a parent Subscriber, the Distributor checks its routing hierarchy for that parent Subscriber:
 - a. If the parent Subscriber is in the routing hierarchy, the Distributor uses that route to send the Distribution to the Subscriber.
 - b. If the parent Subscriber is not in the routing hierarchy, the Distributor sends the Distribution directly to the parent Subscriber of the end-node target Subscriber.
4. If the Subscriber does not have a parent Subscriber, the Distributor checks its routing hierarchy for the Subscriber:

- a. If the Subscriber is in the routing hierarchy, the Distributor uses that route to send the Distribution to the Subscriber.
- b. If the Subscriber is not in the routing hierarchy, the Distributor sends the Distribution directly to the Subscriber.

In other words, if the Distributor can find a way to send the Distribution using its routing hierarchy, it will use the path in that hierarchy to get the Distribution to the Subscriber. Otherwise, it will send the Distribution directly to the Subscriber (or its parent Subscriber).

For that reason, you should make sure every Subscriber that regularly receives Distributions from a Distributor have some connection to the Distributor's routing hierarchy. This connection can be made by being listed in the hierarchy or by having one of the Subscribers in the hierarchy be its parent Subscriber.

You should generally not allow the Distributor to send Distributions over WAN links, except to such Subscribers that may be in the first tier of its routing hierarchy.

Consider the following in designing your Distributor's routing hierarchy:

- ◆ **End-Node Subscribers:** The only Subscribers that you need to add to the routing hierarchy are those you want to be used to pass on Distributions. End-node Subscribers that will only receive Distributions and not pass them on do not need to be added to the routing hierarchy.
- ◆ **Configuring Distribution Routes:** To create the distribution routes, consider your network design and the number of Subscribers on each LAN. Then design the routing hierarchy to mimic your network topology.
- ◆ **Selecting Multiple Subscribers:** During hierarchy creation, you can place multiple Subscribers at the same tier under a single Distributor or Subscriber.
IMPORTANT: The most efficient routing hierarchy is to have more tiers and fewer Subscribers per tier, than just a few tiers with many Subscribers per tier. Therefore, only select a few Subscriber servers per tier. This minimizes the workload for the Distributor or Subscriber server that is sending Distributions to other Subscriber servers. Tiering helps to share the workload of sending Distributions throughout the network.
- ◆ **Using Multiple Distributors:** Multiple Distributors can use the same routing hierarchy of Subscribers, so that the same distribution route can be used by each Distributor.
- ◆ **Reusing Subscribers:** You should consider whether you might overload a Subscriber server if it should be a parent Subscriber in a routing hierarchy that services multiple Distributors.

Selecting Subscribers for the Distribution Routes

The purpose of the Distributor's routing hierarchy is to create the most efficient method for distributing to Subscribers. You need to determine which servers are best suited to be Subscribers in a routing hierarchy, and how many servers to include in the hierarchy.

Select a server that is robust in its physical configuration. For example, a fast CPU, plenty of RAM, and plenty of free hard disk space (especially on volumes other than SYS: on NetWare servers).

Use the following criteria to determine which Subscribers to include in a Distributor's routing hierarchy:

- ◆ Is the Subscriber needed to minimize the Distributor's workload?
- ◆ Do you need other Subscribers to share the workload of a parent Subscriber on a given LAN?
- ◆ Is the Subscriber needed to minimize network traffic (such as through WANs or firewalls)?

To identify the Subscriber servers that will be used in a Distributor's routing hierarchy, create a list of the servers in your network that you want to use as parent Subscribers in a Distributor's routing hierarchy.

To help minimize network traffic, select at least one server on each LAN.

Identify the server objects that can be parent Subscribers in the Distributors' routing hierarchies:

CONFIGURATION PLANNING WORKSHEET

Under **item 16**, enter the names (including full context) for your parent Subscriber servers.

Configuring the Distribution Routes

Enter the following information on your network diagram:

CONFIGURATION PLANNING DIAGRAM

Write "parent=1" next to every location on the diagram that is separated from the Distributor's location by a WAN link or firewall (unless there is only one Subscriber at that location).

Enter the following information on your network diagram:

CONFIGURATION PLANNING DIAGRAM

For every location on the diagram that requires additional parent Subscribers because of the high number of Subscribers, change "parent=1" to "parent=#" where # is the number of parent Subscribers the site will need for load-balancing.

Also note whether you want to use one parent Subscriber in a given location as the primary parent Subscriber (the only one at that location in the Distributor's routing hierarchy) for receiving Distributions and passing them on to other parent Subscribers in that location.

Be sure to include parent Subscribers at the Distributor's location, if needed.

Using the information from your network diagram, design your Distributors' routing hierarchies using the Subscribers you have selected:

CONFIGURATION PLANNING WORKSHEET

Under **item 15**, create a hierarchy for each Distributor's routing hierarchy. You can reuse Subscriber servers in different Distributor's hierarchies.

Understanding Distribution Security

ZfS provides adequate security for Distributions that are sent within a secured network using certificates. However, Distributions could require additional security measures that are available in ZfS.

For more information about security, see [Chapter 20, "Security in Policy and Distribution Services," on page 549](#).

Review the following to determine whether you need any additional security for your Distributions:

- ◆ [“Determining Whether You Need Inter-Server Communications Security” on page 332](#)
- ◆ [“Determining Whether You Need Encryption Security for Windows Servers” on page 332](#)

Determining Whether You Need Inter-Server Communications Security

Policy and Distribution Services uses XMLRPC (Extensible Markup Language Remote Procedure Call) for its normal inter-server communications. XMLRPC optionally provides security for inter-server communication that can be used for communicating securely across non-secured connections.

Policy and Distribution Services can use this security for inter-server communications between servers across non-secured connections, or between a management workstation and servers across non-secured connections. For example, firewalls, intranets, NAT configurations, and so on.

This inter-server communications security ensures that data received across a non-secured connection is from a trusted source, that it has not been tampered with en route, and that the data received can be trusted by other machines. This is accomplished through the use of signed security certificates and digital signatures.

This security requires modifications to certain text files, and is installed using a Zfs wizard.

The following are instances when you would want inter-server communication security:

- ◆ **ConsoleOne Administration:** When you use a workstation to manage a Distributor server across a non-secured connection.
- ◆ **SET Parameters:** When you create a SET Parameter policy or a software package for SET parameters, inter-server communication takes place to provide the target server’s SET parameter information. This communication could cross a non-secured connection.
- ◆ **Server Down Policy:** When you use this policy to down a server, the communication between the downed server and another server watching for it to come back up could cross a non-secured connection.

For more information, see [“Security for Inter-Server Communication Across Non-Secured Connections” on page 562](#).

CONFIGURATION PLANNING WORKSHEET

Under [item 13](#), enter the NetWare and Windows servers where you need to install the inter-server communications security software.

Determining Whether You Need Encryption Security for Windows Servers

You normally do not need to encrypt Distributions that are sent within your secured network. However, you can use encryption to provide security for when you send Distributions outside your network. The NICI software is used for encrypting Distributions.

For NetWare servers, NICI is automatically installed. Therefore, you do not need to do any setup to use Distribution encryption for NetWare servers.

For Windows, Linux, and Solaris servers, you must install NICI on the Distributor and Subscriber servers where you expect encrypted Distributions to be built and extracted.

If you need to install the NICI software on a Windows, Linux, and Solaris server, you must also install that same version on all Distributor and Subscriber servers in your network. Encryption will not work correctly if there are two different versions of NICI installed in your network.

For information on Distribution encryption, see “[Distribution Security Using Encryption](#)” on [page 558](#).

CONFIGURATION PLANNING WORKSHEET

Under [item 14](#), enter the Windows, Linux, and Solaris servers where you need to install the NICI software.

Determining the Channels for the Distributions

Channels are used to group Distributions, to establish a schedule for passing a Distributor’s Distributions on to Subscribers, and to list the Subscriber that are subscribed to the Channel so that the Distributor will know where to physically send the Distribution files.

A Channel can be created for a specific type of Distribution (such as virus pattern files, operating system support packs, or policy packages), or for a specific Distribution time (such as off peak Distributions).

A Channel can be associated with Distributions from many Distributors. A Channel can be subscribed to by many Subscribers.

Subscribers subscribe to Channels in order to receive certain Distributions. Distributors associate their Distributions with the Channels so that the subscribed Subscribers can receive those Distributions.

If you are installing multiple Distributors, they can share Channels for their Distributions. For example, if Distributor A and Distributor B both want to send some of their Distributions to the same set of Subscribers, one Channel can be used by both Distributors.

Channels are used in providing Distributions to Subscribers. Consider the following:

- ◆ A Channel is not owned by any particular Distributor
- ◆ Distributors associate their Distributions with the Channels
- ◆ A Channel can have Distributions from multiple Distributors
- ◆ A Channel can be used to group related Distributions
- ◆ A Channel’s schedule determines when the listed Distributions will be sent
- ◆ A Subscriber subscribes to one or more Channels to receive all of the Distributions listed in those Channels
- ◆ A Subscriber cannot select an individual Distribution from the several that could be listed in a Channel (it must receive all of the Channel’s Distributions)

In naming Channels, use a descriptive method. For example:

```
VirusProtect  
VProtectPatterns  
VirusProtection  
NW51patch4  
NW6patch1  
AUTOEXECNCF000326
```

You will be able to manage your Channels more easily by:

- ◆ Using names that are purpose oriented
- ◆ Using a similar name for the Channel and its Distributions

CONFIGURATION PLANNING WORKSHEET

Under [item 21](#), enter your Channel names. Make the names unique to help identify which Distributions they will send.

You would generally create a Channel for one or more related Distributions. However, for distribution flexibility, you could create one Channel for each application to be distributed.

CONFIGURATION PLANNING WORKSHEET

For each Channel, under [item 22](#) enter the Distributions that belong to the Channel.

For ease of management, plan to create the Channel objects in the same context as your other TED objects, especially the Distribution objects.

CONFIGURATION PLANNING WORKSHEET

Under [item 20](#), enter the eDirectory context where the Channel object should be created.

Determining Subscribers' Subscriptions

You need to subscribe your Subscribers to Channels before they can receive their Distributions. This is done by subscribing a Subscriber or Subscriber Group to the Channel that is associated with the Distribution it needs:

- ◆ [“Subscribers” on page 334](#)
- ◆ [“Subscriber Groups” on page 334](#)

Subscribers

Because Subscribers do not access eDirectory, all configuration information in the Subscriber object's properties is pushed down to it from the configuring Distributor, if it is needed. This includes such information as working directory, log file level and location, console messaging level, variables, and so on.

Changes to a Subscriber object's properties are not in effect until the Distributor re-reads eDirectory and sends a new Distribution with the configuration information down to the Subscriber.

For each Distribution, determine which Subscriber servers will need a particular Distribution.

CONFIGURATION PLANNING WORKSHEET

Under [item 24](#), enter the Channel name for a Distribution (see [item 22](#)) and list the Subscribers that need that Distribution. Repeat for each Channel you entered in [item 21](#).

Subscriber Groups

A Subscriber Group is used for grouping Subscribers that have the same Distribution needs.

Subscriber Groups are useful when you will be sending several different Distributions to the same set of Subscribers. There is no need to create a Subscriber Group if it will only be associated with one Channel.

For example, Distribution A will be in Channel A, Distribution B will be in Channel B, and so on. Then, without using a Subscriber Group, you would need to subscribe each of your Subscribers to Channel A, then each to Channel B, and so on, which could be a very long process. However, by using a Subscriber Group, you will only need to create the group, add the Subscribers to it, then subscribe that one group to each Channel.

Another use of a Subscriber Group is that when the group is associated with two or more Channels, you can edit the group's membership more easily than making the same changes in multiple Channels. For example, to remove a Subscriber from one Subscriber Group, you just edit that one group's properties. To remove that same Subscriber from several Channels, you would need to edit each Channel's properties.

CONFIGURATION PLANNING WORKSHEET

Under [item 17](#), enter a unique name for the Subscriber Group.

Under [item 18](#), enter a list of Subscribers that need the same Distributions from the Channel (see [item 21](#) and [item 22](#)) where the group will be subscribed.

Under [item 24](#), enter the Channel names for the Distributions that you want all of the Subscribers in the group to receive.

Determining the Distribution Schedules

TED has different schedules so that you can coordinate the various distribution processes. Review the following to plan your TED schedules:

- ◆ [“Understanding Scheduling in TED” on page 335](#)
- ◆ [“Determining the Distributors’ Refresh Schedule” on page 336](#)
- ◆ [“Determining the Distribution’s Build Schedule” on page 336](#)
- ◆ [“Determining the Channels’ Send Schedules” on page 336](#)
- ◆ [“Determining the Subscribers’ Extract Schedules” on page 336](#)

Understanding Scheduling in TED

Both TED objects and individual Server Policies can be scheduled.

TED uses schedules to control when Distributors are refreshed and Distributions are built, sent, and extracted. Schedules do not affect the total resources used by a Distribution, but rather *when* the resources will be used.

Some policies must be scheduled before they can be enforced. If you enable a policy, but do not schedule it, it will be activated according to the schedule currently specified in the Default Package Schedule, which provides a default for scheduled policies. The default schedule is Run At System Startup.

If you configure several policies with the same schedule, the order they are run depends on the time stamps created when you created the policies. Therefore, when you view a list of policies, the order they are listed is the order that they will be run.

If you want to control the order that certain policies are run, you should stagger their schedules, rather than rely on the time stamps to determine when they will run. Therefore, consider the TED schedules you select when scheduling your policies, so that you do not have undesirable overlap, or out-of-sequence events that could cause some scheduled items to fail.

Other issues you may need to understand:

- ◆ How time zones can affect scheduling
- ◆ How policy schedules are affected by distribution schedules
- ◆ How distribution schedules can be affected by Distributor and Subscriber servers' non-ZfS software usage
- ◆ How the Randomly Dispatch option can affect scheduling
- ◆ How the Active and Inactive object options for the TED objects can affect scheduling and distribution flow

For more information, see [Chapter 21, "Scheduling," on page 565](#).

Determining the Distributors' Refresh Schedule

The Refresh schedule determines when the Distributor will re-read eDirectory for configuration changes.

This enables the Distributor to respond to a request to build a Distribution. The Distributor rebuilds a Distribution when it discovers that there are configuration changes within eDirectory.

You will also be instructed to manually refresh your Distributors to start the distribution process, because that schedule is set to Never by default. You can change this schedule later after you have reviewed and understood [Chapter 21, "Scheduling," on page 565](#).

Determining the Distribution's Build Schedule

The Build schedule determines when a Distributor will be requested to build the individual pieces that comprise the Distribution.

During configuration, you will be instructed to set each Distribution's Build schedule to allow the Distribution to be sent immediately after building it.

Determining the Channels' Send Schedules

The Send schedule provides a window of time for when a Distributor can send its Distributions to the Subscribers.

During configuration, you will set each Channel's Send schedule to an interval of every 5 minutes, meaning that the Distributor can send its Distributions at any of the 5-minute intervals when the Channel's schedule fires.

Determining the Subscribers' Extract Schedules

The Extract schedule determines when a Subscriber can start to extract a Distribution that has been received.

Before a Subscriber can use a Distribution that is sent to it, it must first extract the Distribution. Therefore, the Subscriber's Extract schedule should be set before you send the Distributions.

Determine when you want the various Subscriber servers to be active extracting Distributions. Depending on a Distribution's size, it could be best to have Distributions extracted during off-peak hours. For information on scheduling issues involving time zones, see [“Scheduling Issues” on page 565](#), especially [“Calculating Time Differences” on page 568](#).

CONFIGURATION PLANNING WORKSHEET

Under [item 23](#), enter the Subscribers' extract schedules.

Configuring Your Distribution System

Use these sections in the following order:

1. [“Installing Additional Distributors, Databases, and Subscribers” on page 337](#)
2. [“Setting Up Distributors in a Mixed Network Operating System Environment” on page 340](#)
3. [“Setting Up Additional Distribution Security” on page 340](#)
4. [“Starting the Distributor Agents” on page 341](#)
5. [“Setting Up the Additional Databases” on page 343](#)
6. [“Configuring the Distribution Flow” on page 344](#)
7. [“Creating the Distributions and Related Channels” on page 345](#)
8. [“Subscribing to the Distributions” on page 347](#)
9. [“Sending the Distributions” on page 348](#)

Installing Additional Distributors, Databases, and Subscribers

When installing Policy and Distribution Services for the first time, you installed one Distributor with a database file. If you planned to install more Distributors or databases (see [“Understanding Distributors” on page 379](#) and [“Determining How Many Databases You Need” on page 589](#)), you should perform this installation now.

When installing Policy and Distribution Services for the first time, you might not have installed the Subscriber software to all of your servers. If you want to install the Subscriber software to more servers at this time, you should perform this installation now.

IMPORTANT: Any servers where you do not have the Subscriber software installed will not be eligible to receive the Distributions you have planned to create and distribute at this time. However, when you install the Subscriber software to servers at a later date, they can be subscribed to existing Channels for receiving its Distributions.

To install additional Distributors, and databases, and Subscriber software to more servers, do the following in order:

1. [“Preparing to Install” on page 338](#)
2. [“Starting the Installation Program” on page 338](#)
3. [“Selecting and Configuring the Distributor and Subscriber Servers” on page 338](#)
4. [“Completing the Installation” on page 339](#)

Preparing to Install

- 1 Make sure you have fulfilled all of the necessary requirements for your target Distributor and Subscriber servers.
- 2 If Java has not been unloaded on the target NetWare servers, unload JAVA.NLM.

For example, at each NetWare server's console prompt, enter:

```
java -exit
```

- 3 Select the workstation you will use to install the ZfS Distributors and Subscribers.
- 4 If you have not already done so, log in to the eDirectory tree where you will be creating the ZfS objects (worksheet [item 1](#)).

This should be the same tree where you extended the schema for ZfS 3.0.2.

You will automatically be authenticated to all of the NetWare target servers in this tree during installation. You will be able to select those servers, as well as servers in other trees or domains, for installing the Policy and Distribution Services software. However, this is the tree where all of the ZfS objects will be installed for each of the selected servers.

Starting the Installation Program

- 1 On the installation workstation, insert the *ZENworks for Servers Program* CD or the *ZENworks 6 Server Management Program* CD.

The startup screen is displayed. If the startup screen is not automatically displayed after inserting the CD, you can start it by running WINSETUP.EXE at the root of the CD.

IMPORTANT: Installation from a CD in a remote server is not supported unless there is a drive mapped on the workstation to that remote server. For example, if you place the CD in a Windows NT/2000 server CD drive, then run the installation from a workstation, you must have a drive mapped on the workstation to the CD drive of that NT/2000 server.

- 2 Click the Policy-Enabled Server Management option.
This begins the installation program.
- 3 If you agree with the Software License Agreement, click Accept > Next.
- 4 On the Installation Type page, click New Installation > click Next.
- 5 On the Components to Install page, click the Tiered Electronic Distribution, Server Policies, and Server Software Packages check box > click Next.
- 6 On the Installation Options page, make sure both the Create and Install check boxes are checked.
- 7 On the eDirectory Tree for Creating Objects page, select the tree (worksheet [item 1](#)).

This is the tree where you initially created ZfS objects.

Selecting and Configuring the Distributor and Subscriber Servers

- 1 On the Server Selection page, click Add Server > browse for the Distributor (worksheet [item 2](#)) and Subscriber (worksheet [item 3](#)) servers > click OK.
- 2 For each Distributor server, click the check box in the Distributor column.
Uncheck the box under the Subscriber/Policies column only if you are sure you do not want the Subscriber and Server Policies software installed on that Distributor server.
- 3 For each Subscriber server, click the check box in the Subscriber/Policies column.

- 4** If you plan to install a database on a Distributor server (worksheet [item 4](#)), for one of the servers, click the check box in the Database column > click Next.

You can install only one database per run of the installation program. Therefore, click the Database column for just one of the Distributors.
- 5** On the Installation Paths and Options page, for each Distributor server, edit the installation path if you do not want to use the default (worksheet [item 5](#)).

If you want all Distributor servers to have the same installation path, select all of the servers, then edit the path.
- 6** For each Subscriber server, edit the installation path if you do not want to use the default (worksheet [item 6](#)).

If you want all Subscriber servers to have the same installation path, select all of the servers, then edit the path.
- 7** To launch Policy and Distribution Services components on server startup, click the check box.
- 8** On the Distributor Object Properties page, edit the properties as necessary (worksheet [item 7](#)) > click Next.
- 9** On the Subscriber Object Properties page, edit the properties as necessary (worksheet [item 8](#)) > click Next.
- 10** For the Distributor server where you selected to install the database, do the following:
 - ◆ Edit the database file's path if you do not want to use the default (worksheet [item 9](#)).

Because the database file can become very large, we recommend that you change the default NetWare volume from SYS: to another volume on that server.
 - ◆ Edit the Database object's name, if desired (worksheet [item 10](#)).
 - ◆ Change the Database object's container, if desired (worksheet [item 11](#)).
- 11** Click Next.

The Summary page is displayed.

Completing the Installation

- 1** To save the current installation configuration for future use in installing Distributors, on the Summary page click the Save the Following check box > enter a path and filename for the template file.

If you attempt to quit the installation program without clicking Finish, you will be prompted to save your current installation configuration to an installation template file.

You can reuse this template to speed up filling in installation pages in subsequent installations of Distributors or Subscribers.
- 2** Click Finish to begin the installation process.
- 3** After the installation program has finished, review the installation log file to determine whether any components failed to install.

The log file is located at:

```
C:\TEMP\_RESNumber.TXT
```

where *Number* is increased incrementally each time a new installation log is created.
- 4** If necessary, rerun the installation program.

Select only the components that failed to install.

- 5 Rerun the installation program once for each additional database that needs to be installed (worksheet [item 4](#)).

On the Server Selection page add only one of the Distributors where you planned to have a database installed, but have not installed it yet. Then, click only the Database column for that database's Distributor server and fill in the applicable information on the remaining installation pages.

Setting Up Distributors in a Mixed Network Operating System Environment

In ZFS 3.0.2, Distributor servers must be able to authenticate to the eDirectory 8.x tree. If your network has both eDirectory 8.x and NDS 7.x installed, you must edit the TED.NCF file on each of your NetWare Distributor servers (worksheet [item 2](#)) to ensure that they can authenticate to an eDirectory 8.x tree.

To edit the TED.NCF files:

- 1 On a Distributor server's file system, open SYS:\ZENWORKS\PDS\TED\TED.NCF in a text editor.

The path to your ZENWORKS directory might be different if you used a different volume or inserted other path information between the volume and the ZENWORKS directory.

- 2 Locate the line similar to the following (usually at the end of the file):

```
java -mx128M -envDISPLAY=127.0.0.1:0 -noclassgc -ns -jstedexit -snTed -  
classpath $tedpath com.novell.application.zenworks.ted.TED CORPTREE  
"Distributor_Server001.TED.ZENworks.Novell" distributor_password
```

- 3 Locate the server's tree name (usually immediately after the com.novell.application.zenworks.ted.TED phrase) > replace it with the IP address of a server that has eDirectory 8.x installed (worksheet [item 12](#)).

The IP address can be from the Distributor server where you are editing the TED.NCF file, or the IP address of any other server running eDirectory 8.x.

The line should now appear as:

```
java -mx128M -envDISPLAY=127.0.0.1:0 -noclassgc -ns -jstedexit -snTed -  
classpath $tedpath com.novell.application.zenworks.ted.TED 155.55.155.55  
"Distributor_Server001.TED.ZENworks.Novell" distributor_password
```

- 4 Save the configuration file > exit the text editor.

Setting Up Additional Distribution Security

To ensure that you have the proper security for your Distributions, do the following tasks that are applicable:

- ◆ [“Setting Up Inter-Server Communications Security” on page 340](#)
- ◆ [“Installing NICI 2.4” on page 341](#)

Setting Up Inter-Server Communications Security

If you will be distributing to servers outside your secured network (worksheet [item 13](#)), see [“Security for Inter-Server Communication Across Non-Secured Connections” on page 562](#) for detailed instructions on setting up security for inter-server communications.

Installing NCI 2.4

If you need Distribution encryption support for certain NetWare, Windows, Linux, and Solaris Subscriber servers, a newer version of NCI (2.4) provides this support. A NCI update is contained on the ZENworks *Companion* CDs. The NCI24CPK.EXE file on the CD is a self-extracting file that contains the NCI24.CPK software package file.

This software package updates NCI to the 2.4 version. Because NetWare 5.1/6 servers automatically have this version of NCI installed, you only need to install the NCI24.CPK software package to the Windows, Linux, and Solaris Subscriber servers where you are using the encryption feature of TED.

IMPORTANT: All servers that will be sending or receiving encrypted Distributions must be running the same version of NCI. Otherwise, encrypted Distributions to any of those servers will fail.

When you install NCI24.CPK, it will not check to see if NCI is already installed. It will simply install NCI to all Subscribers subscribed to the Channel that you select for the software package used to distribute NCI.

The NCI24.CPK file is the same software package file that was provided with ZfS 3 SP1. If you previously updated your servers to NCI 2.4 using SP1, you can skip this section.

- 1 On a Windows workstation, insert the *ZENworks for Servers Companion* CD or the *ZENworks 6 Companion 1* CD.
- 2 Copy the NCI24CPK.EXE file from a *Companion* CD (`CD_drive:\NICI` or `CD_drive:\ZENWORKSFORSERVERS\nICI`) to a directory on the Windows workstation.
- 3 From the directory where you saved the NCI24CPK.EXE file, run this .EXE file to extract the NCI24.CPK and README_NICI24CPK.TXT files.
- 4 Follow the installation instructions in the Readme file under "Installing NCI 2.4 with ZfS 3.0.2."

In Step 6 of the Readme, select the servers that you planned to update to NCI 2.4 (worksheet [item 14](#)).

Starting the Distributor Agents

Before using ConsoleOne to further configure Policy and Distribution Services, you need to start the agents.

- ◆ [“Starting the Agents” on page 341](#)
- ◆ [“Verifying That the Policy and Distribution Services Agents Are Loaded” on page 342](#)

Starting the Agents

- 1 On a NetWare server where you installed the software for the Distributor (worksheet [item 2](#)) or Subscriber (worksheet [item 3](#)), at the server’s console prompt, enter:

```
sys:\zenworks\pds\smmanager\zfs.ncf
```

If you used a different volume, or added other path information before the ZENWORKS directory, replace the SYS: portion with the alternate path information.

After you have started ZfS in this manner, and after the server has rebooted once, the full path will no longer be needed for start the software—you will only need to enter zfs thereafter. By entering the path the first time you run ZFS.NCF, or by rebooting the server after installing ZfS, you enable the server to learn that path.

IMPORTANT: If you edited the TED.NCF file for a Distributor that already has ZfS running (as instructed in “[Configuring Distributors in a Mixed eDirectory Environment](#)” on page 328), bring the ZfS Agent down and restart it on that server.

TED.NCF and ZWS.NCF are started automatically by the ZFS command. The database is automatically started by the installation program.

- 2** Repeat [Step 1](#) for each NetWare server in your network where you have installed the Distributor or Subscriber software.
- 3** On a Windows server where you installed the software for the Distributor (worksheet [item 2](#)) or Subscriber (worksheet [item 3](#)), do the following:
 - 3a** Open the Control Panel.
 - 3b** Do the applicable tasks:
 - On Windows NT, double-click Services.
 - or
 - On Windows 2000, double-click Admin Tools > double-click Services.
 - 3c** Start the Novell ZfS Policies service.

This will also start the Novell ZfS Distribution service, and the Novell ZfS Web Server service. The Novell Sybase* Database service is automatically started by the installation program.
- 4** Repeat [Step 3](#) for each Windows server in your network where you have installed the Distributor or Subscriber software.
- 5** At the server console or in an Xterm window on a Linux or Solaris server where you installed the software for the Distributor (worksheet [item 2](#)) or Subscriber (worksheet [item 3](#)), enter:

```
/etc/init.d/zfs start
```
- 6** Repeat [Step 5](#) for each Linux or Solaris server in your network where you have installed the Distributor or Subscriber software.

Verifying That the Policy and Distribution Services Agents Are Loaded

To verify that the Policy and Distribution Services agents are running on the target servers:

- ◆ “[Verifying on NetWare Servers](#)” on page 342
- ◆ “[Verifying on Windows Servers](#)” on page 343
- ◆ “[Verifying on Linux or Solaris Servers](#)” on page 343

Verifying on NetWare Servers

To verify if ZfS is running properly on a NetWare server:

- 1** On the target server’s console, press Ctrl+Esc to view the loaded software programs.
- 2** If the ZfS item (Policy/Package Agent) is not displayed, review the ZFSINIT.TXT file (under ZENWORKS\PDS\SMANAGER), which contains information about why the agent did not start.

Use this information to solve the problem.

This file is used to log only startup problems.

- 3** If the TED item (TED Agent) is not displayed, review the DEFAULTLOG.TXT file (under PDS\TED), which contains information about why the agent did not start.
Use this information to solve the problem.
This file is used to log only startup problems.
- 4** Repeat **Step 1** through **Step 3** for each NetWare server.
- 5** If necessary, rerun the installation program.

Verifying on Windows Servers

To verify if ZfS is running properly on a Windows server:

- 1** On the target server, open the Control Panel > double-click Services (on Windows 2000, double-click Admin Tools > click Services) > determine if the following services are running:
 - Novell ZfS Policies
 - Novell ZfS Distribution
 - Novell Zen Web Server
 - Novell Sybase Database
- 2** Repeat **Step 1** for each Windows server.
- 3** If necessary, rerun the installation program.

Verifying on Linux or Solaris Servers

To verify if ZfS is running properly on a Linux or Solaris server:

- 1** At the server console or in an Xterm window on a Linux or Solaris server, enter:

```
/etc/init.d/zfs status
```
- 2** If the TED agents (Tiered Electronic Distribution component) do not start, check the defaultLog.txt file in the ted directory.
- 3** If the Policy/Package Agent (ZfS policies component) does not start, check the ZFSINIT.LOG file in the smanager directory.
- 4** To look up agent startup errors, see “**Tiered Electronic Distribution Errors**” and “**Policy/Package Agent Errors**” in “**Policy and Distribution Services**” in the *Troubleshooting* guide.
Resolve the problem, then start the agents successfully.

Setting Up the Additional Databases

If you installed additional ZENworks databases, you should do the following:

- 1** In ConsoleOne, right-click a ZENworks Database object (worksheet **item 10**) > click Properties.
- 2** On the ZENworks Database tab, click either the Server DN or Server IP Address radio button.
One of these location IDs might already be the default. If not, enter the information, which should be for the server where ZFSLOG.DB resides.
- 3** Click the eDirectory Rights tab > Trustees of This Object > Add Trustee > select [Public].
The database object must be assigned a trustee of Public, or the Policy/Package Agent will display messages that it cannot connect with the database or read the ZENworks for Servers policy.

4 Click OK.

If you click Cancel, none of the information you added or changed on any of the tabs will be saved. However, the database object will remain on the tree.

5 Set up the ZENworks Database policy.

For steps to specify the location of a database, see [“ZENworks Database” on page 484](#).

6 Associate the Service Location Package with a container above where the Distributor object resides.

7 Repeat [Step 1](#) through [Step 6](#) for each new Database object that you installed.

Configuring the Distribution Flow

You need to configure your distribution system to ensure the most efficient use of your network in sending Distributions by setting up the Distributors' routing hierarchies. This was not done for any Distributor when you installed Policy and Distribution Services.

To configure your distribution system:

- ◆ [“Configuring the Distributor Routing Hierarchies” on page 344](#)
- ◆ [“Configuring Parent Subscribers” on page 345](#)
- ◆ [“Configuring Subscriber Groups” on page 345](#)

Configuring the Distributor Routing Hierarchies

1 In ConsoleOne, right-click a Distributor object (worksheet [item 2](#)) > click Properties.

2 Click the Routing tab > do the following:

2a Click Add > browse for your first tier Subscriber servers (worksheet [item 15](#)) > click Select > click OK.

This sets up your first tier of Subscriber servers. These will receive Distributions directly from the Distributor.

2b Click one of the Subscriber servers in the first tier of the routing tree > click Add > browse for your next tier of Subscriber servers to go under that first tier Subscriber (worksheet [item 15](#)) > click Select > click OK.

This sets up a second tier of Subscriber servers for the one Subscriber that you selected. These second-tier Subscribers will receive Distributions indirectly from the Distributor via the Subscriber server above them in the hierarchy.

2c Repeat [Step 2b](#) for each of the first-tier Subscribers until you have selected all of the second-tier Subscribers for this part of the hierarchy.

2d Click one of the Subscriber servers in the second tier of the routing tree > click Add > browse for your next tier of Subscriber servers to go under that Subscriber (worksheet [item 15](#)) > click Select > click OK.

2e Repeat [Step 2d](#) for each of the second tier Subscribers until you have selected all of the third-tier Subscribers for this part of the hierarchy.

2f Repeat this process, tier by tier, until you have completed your planned routing hierarchy for the current Distributor.

3 Repeat [Step 1](#) through [Step 2](#) for your other Distributors.

- 4 When you have finished building the routing hierarchy, click OK.

Configuring Parent Subscribers

All Subscribers should not receive their Distributions directly from a Distributor. The Distributor's routing hierarchy provides a way to minimize the Distributor's workload in sending Distributions.

For Subscriber servers to receive their Distributions using the routing hierarchy, you need to identify a parent Subscriber that is in the routing hierarchy for each end-node Subscriber (the Subscriber to receive the Distribution). This will allow an end-node Subscriber to receive its Distributions through the routing hierarchy, rather than directly from a Distributor.

A Subscriber that is in the Distributor's routing hierarchy does not need to have a parent Subscriber in order to receive a Distribution from that Distributor. Distributors check their routing hierarchies first, then check for parent Subscribers second.

To associate Subscribers with parent Subscribers:

- 1 In ConsoleOne, select a group of Subscriber objects for servers that you planned to have serviced by a particular parent Subscriber (worksheet [item 16](#)) > right-click the selected group > click Properties of Multiple Objects > in the Parent Subscriber field, browse for the parent Subscriber object > click OK > OK.

Because you can do multiple editing of eDirectory objects, you can select all of the Subscribers that will be serviced by one parent Subscriber and edit the Parent Subscriber field once for all of them.

- 2 Repeat this process for all end-node Subscribers.

Configuring Subscriber Groups

To create and populate a Subscriber Group:

- 1 In ConsoleOne, select the container to hold the Subscriber Group object > click File > New > Object > TED Subscriber Group.
- 2 In the New TED Subscriber Group dialog box, enter a name for the Subscriber Group (worksheet [item 17](#)) > click Define Additional Properties > click OK.
- 3 In the General Settings tab, enter a description.
- 4 To populate the group with Subscribers, click the Members tab > do the following:
 - 4a Click Add > browse for and select the Subscriber objects (worksheet [item 18](#)) > click OK.
 - 4b To remove any Subscribers from the list, select the Subscribers > click Delete.
 - 4c To view the properties of any Subscriber, select the Subscriber > click Details.
- 5 Click OK when you have finished configuring the Subscriber Group object.

Creating the Distributions and Related Channels

The following are generic instructions for creating a Distribution. For more detailed instructions for most Distribution types, see "[Tiered Electronic Distribution](#)" on page 371. For steps on using the Distribution Wizard to create a File or FTP Distribution, see "[Using the TED Distribution Wizard](#)" on page 414.

For your initial deployment of Policy and Distribution Services, you created a Distribution using the Policy Package type for the required distributed policies (see "[Setting Up the Necessary Server](#)").

Policies” under “Installing on NetWare and Windows Servers” in “Installing Policy and Distribution Services on NetWare and Windows Servers” in the *Installation* guide). At this time, you can create Distributions for other policies that you have planned.

You first need to create the Distribution, then create the Channel (if you don’t use an existing Channel):

- ◆ “Creating and Configuring the Distribution” on page 346
- ◆ “Creating and Configuring the Channel” on page 347

Creating and Configuring the Distribution

- 1** In ConsoleOne, locate the container where the TED objects were installed.
- 2** Right-click the container > click New > Object > select TED Distribution.
- 3** Enter a Distribution name (worksheet [item 19](#)).
Name the Distribution so you can identify what it contains.
- 4** Browse to the Distributor object that will own this Distribution (worksheet [item 19](#)) > select it.
Each Distribution is associated with a single Distributor. That Distributor is responsible for building and sending the Distribution.
- 5** Click the Define Additional Properties check box.
- 6** Click OK to create the object.
The properties for the Distribution are now displayed.
- 7** Click the Type tab > in the Select Type drop-down box, click a Distribution type (worksheet [item 19](#)).
- 8** Configure the Distribution.
For information on configuring the different Distribution types, see “Distributions” on [page 393](#).
Use the up and down arrow buttons to change the distribution order.
- 9** Click the Schedule tab.
The Distribution's schedule determines how often the Distributor will attempt to build a new version of the Distribution. A new version is built only if there have been changes since the last version was built.
- 10** Select Run Immediate from the drop-down list.
This will cause the Distributor to build the Distribution as soon as it re-reads eDirectory for the Distribution information.
- 11** Click OK at the bottom of the Distribution Properties dialog box to save all changes.
- 12** If you have not previously resolved certificates, click Yes when prompted to copy security certificates.
For information on resolving certificates, see “Resolving Certificates” on [page 552](#).
The Distributor needs to have been run at least once so that its certificates can be minted (created).
A Distributor needs to resolve its certificates only once per Subscriber.

The Subscriber software does not need to be running on the server for security certificates to be resolved. The server only needs to be up.

ConsoleOne will send security certificates to each Subscriber server that subscribes to the Channel that was selected in the Channel Tab. Each Subscriber must have a security certificate from the Distributor before it can receive Distributions from that Distributor.

It can take several minutes to copy a security certificate to each Subscriber.

IMPORTANT: Certificate copying only needs to be done once for each Distributor/Subscriber relationship.

- 13** If you receive an error when the Distributor tries to copy to an NT Subscriber, enter the following for the path:

```
\\IP_Address\zen$\PDS\TED
```

where *IP_Address* is the IP address of that NT Subscriber.

- 14** If you receive an error when the Distributor tries to copy to a Linux or Solaris Subscriber, or you cannot browse for the Server to select it for resolving certificates, you must map a drive to the server (such as through using Samba) and then repeat resolving certificates.
- 15** Repeat these steps for any other Distributions you want to create at this time (worksheet [item 19](#)).

Creating and Configuring the Channel

Channel objects are used to associate Subscribers with Distributions. When Subscribers subscribe to a Channel, they receive all of the Distributions associated with that Channel. Each Channel has a schedule that determines when the Distributions associated with it are to be sent to the Subscribers.

- 1** In ConsoleOne, locate the container where the TED objects reside (worksheet [item 20](#)).

This container should already exist. It is where your Distributor and Subscriber objects were created.

We suggest for ease of management that you use the same OU for all Channels.

- 2** Right-click the TED container > click New > Channel > OK.

- 3** Enter a name for the Channel (worksheet [item 21](#)) > click OK.

You could name your Channels according to the Distributions you intend for them. For example, Channel - ZfS 3 Support Pack 2.

- 4** Right-click the new Channel object > click Properties.

- 5** Click the Distributions tab > click Add > browse for and select the Distributions for the Channel (worksheet [item 22](#)) > click OK.

This associates the Distributions with the Channel. The Subscribers that are subscribed to this Channel will receive the current Distributions.

- 6** To set the Channel's Send schedule, click the Schedule tab > select Interval > specify the interval as every 5 minutes > click OK.

- 7** Repeat [Step 1](#) through [Step 5](#) for each Channel you have planned (worksheet [item 21](#)).

Subscribing to the Distributions

- ◆ [“Setting Subscribers’ Extract Schedules” on page 348](#)

- ◆ [“Subscribing to the Channels” on page 348](#)

Setting Subscribers’ Extract Schedules

Before a Subscriber can use a Distribution that is sent to it via TED, it must extract the Distribution. Therefore, the Subscriber’s extraction schedule must be set before sending the Distributions.

- 1** In ConsoleOne, right-click the Subscriber object (worksheet [item 23](#)) for a server where you want to set the extraction schedule > > click Properties.
- 2** Click the Schedule tab > click the arrow for the drop-down box > click Run Immediately > click OK.

This will cause the selected Subscriber to extract its Distributions as soon as they are received.

- 3** Repeat [Step 1](#) and [Step 2](#) as necessary until all Subscriber schedules have been set.

Subscribing to the Channels

Subscribers must subscribe to a Channel in order to receive the Distributions associated with that Channel. In the following steps, you will associate all of your Subscribers to the Channels created previously.

- 1** In ConsoleOne, right-click a Channel object (worksheet [item 21](#)) > click Properties.
- 2** Click the Subscribers tab > click Add > browse for each of the Subscriber or Subscriber Group (worksheet [item 24](#)) objects to be subscribed to this Channel > click Select > click OK.
- 3** Click the General tab > make sure the Active check box is checked.
- 4** Click OK to close the Channel object’s properties and save the changes.
- 5** Click No when prompted to copy security certificates.
- 6** Repeat [Step 1](#) through [Step 5](#) for each Channel (worksheet [item 21](#)).

Sending the Distributions

Now that you have installed, created, and configured your Distributors, Subscribers, Channels, and Distributions, you can begin the Distribution process.

Do the following in order:

- 1.** [“Scheduling and Refreshing the Distributor” on page 348](#)
- 2.** [“Verifying That the Distribution Process Was Successful” on page 349](#)

Scheduling and Refreshing the Distributor

- 1** In ConsoleOne, right-click the Distributor object (worksheet [item 2](#)).
- 2** On the Distribution object’s Build Schedule tab, click Send Distribution Immediately After Building.

The Distribution will be sent as soon as it is built, regardless of the Channel’s Send schedule.

- 3** Click Refresh Distributor.

This causes the Distributor to re-read eDirectory and obtain all of the changes that were made in eDirectory.

Building the Distribution will begin immediately (according to the Build schedule you set previously). The Distribution will be sent within five minutes (according to the Send schedule you set previously).

As soon as the Subscribers receive the entire Distribution, they will extract the contents to the Subscriber's working directory that you specified in the Subscriber object's properties.

Verifying That the Distribution Process Was Successful

There are a number of ways you can verify that your Distribution process has worked:

- ♦ **Reporting:** Run a report on the Distribution to see its status. For information on TED reporting, see [Chapter 24, "Reporting," on page 599](#).
- ♦ **Log Files:** Depending on the logging levels you are using, you can review the log files for distribution statuses. Log files (.LOG) can be found in the Distributors' and Subscribers' [working directories](#).
- ♦ **Distribution Files:** Compare the Distribution file on the Distributor's file system (under ZENWORKS\PDS\TED\DIST) with the Subscriber's file system (under ZENWORKS\PDS\TED\SUB\individual_Distribution's_path) to see if it was received. The Distribution file uses the same name on both servers.

Managing Your Distribution System

Your Policy and Distribution Services system is now set up and ready for use. You can revisit ["Configuring Your Distribution System" on page 337](#) at any time and use the applicable sections to update your distribution system.

You can manage your distribution system using the ConsoleOne and iManager tools. There is some functionality in one tool that is not in the other. Generally, you can use ConsoleOne for installation and setup tasks, and iManager for management tasks. For more information, see ["Comparing the ZfS Management Role in iManager with ConsoleOne Capabilities" on page 368](#).

For information on using ConsoleOne, see the following:

- ♦ [Chapter 16, "Tiered Electronic Distribution," on page 371](#)
- ♦ [Chapter 17, "Server Policies," on page 455](#)
- ♦ [Chapter 18, "Server Software Packages," on page 493](#)
- ♦ [Chapter 19, "Desktop Application Distribution," on page 525](#)
- ♦ [Chapter 24, "Reporting," on page 599](#)

For information on using iManager, see ["Novell iManager" on page 359](#).

Configuration Planning Worksheet

Use the following worksheet to log configuration information as you plan how to set up your distribution system. You might need to attach lists for some items.

This worksheet is designed to print best from the PDF version of the documentation.

IMPORTANT: Do not use this planning worksheet by itself to configure Policy and Distribution Services, even if you feel experienced enough to do so. There are some required configuration steps that are not covered in this worksheet, because planning is not needed for those steps. Use the sections under ["Configuring Your](#)

[Distribution System](#) on page 337 as your guide for performing the actual configuration of Policy and Distribution Services.

Configuration Information	Instructions
Installing Additional Distributors, Databases, and Subscribers	If you do not have additional Distributors, databases, or Subscribers to install, skip to worksheet item 12 .
1) Tree for the Distributor and ZENworks Database objects:	Enter the name of the eDirectory tree where you will install the ZfS objects. For more information, see “Understanding Your Network Topology” on page 324.
2) Distributor server names:	Enter the server names for each server that you want to be a Distributor. Distributor servers build and own the Distributions. For more information, see “Determining Distributor Properties” on page 326.
3) Subscriber server names:	Enter the server names for each server that you want to be a Subscriber. Subscriber servers receive and extract the Distributions. For more information, see “Other Subscribers To Be Installed?” on page 328.

Configuration Information	Instructions
4) Database server names:	<p>Enter the server names for each server where you want to install the ZENworks database, which can be installed on NetWare and Windows servers.</p> <p>You can have multiple databases for Policy and Distribution Services, but only one per server.</p> <p>Also enter the purpose for each database, or a Distributor identifier for each database if they will each be used the same way.</p> <p>For more information, see “Determining Whether a Distributor Server Will Host a ZENworks Database” on page 327.</p>
5) Installation paths for Distributors’ software:	<p>Enter the path where you want the Distributor software installed. The default is \ZENWORKS for both NetWare and Windows servers.</p> <p>For more information, see “Determining ZFS Software Installation Paths” on page 326.</p>
6) Installation paths for Subscriber software:	<p>Enter the path where you want the Subscriber software installed. The default is \ZENWORKS for both NetWare and Windows servers.</p> <p>For more information, see “Determining ZFS Software Installation Paths” on page 326.</p>
7) Distributors’ properties, where different than the installation defaults:	<p>Edit the following information for your Distributor servers:</p> <ul style="list-style-type: none"> ◆ Distributor object’s name (the default is <code>Distributor_ServerName</code>) ◆ Distributor’s context (using the TED container) ◆ Distributor server’s working directory <p>For more information, see “Determining Distributor Properties” on page 326.</p>

Configuration Information	Instructions
8) Subscribers' properties, where different than the installation defaults:	<p>Edit the following information for your Subscriber servers:</p> <ul style="list-style-type: none"> ◆ Subscriber object's name (the default is <code>Subscriber_ServerName</code>) ◆ Subscriber context (using the TED container) ◆ Subscriber server's working directory <p>For more information, see "Other Subscribers To Be Installed?" on page 328.</p>
9) Installation paths for ZENworks database software:	<p>Enter the path where you want the ZFSLOG.DB file located. The default is <code>ZENWORKS\DATABASE</code>. For NetWare servers, we recommend not using the <code>SYS:</code> volume because the database file can become very large. We also recommend that you install the database software on a server where the Subscriber software is also installed so that you can use the Database Purge option.</p> <p>For more information, see "Determining ZFS Software Installation Paths" on page 326.</p>
10) Database object name:	<p>Either accept the default names, or provide ones that will help you to identify the databases' purposes.</p> <p>For more information, see "Determining Whether a Distributor Server Will Host a ZENworks Database" on page 327.</p>
11) Database object Container:	<p>We recommend you use the same container where your other TED objects reside.</p> <p>For more information, see "Determining Whether a Distributor Server Will Host a ZENworks Database" on page 327.</p>

Configuration Information	Instructions
Configuring the Distributors for a Mixed eDirectory Environment	
12) IP address of server in eDirectory 8.x:	<p data-bbox="931 157 1469 217">If you do not have a mixed eDirectory environment, skip to worksheet item 13.</p> <p data-bbox="931 243 1469 358">Provide the IP address of a server in the tree using eDirectory 8.x. This can be the Distributor server's IP address, if that server is running eDirectory 8.x.</p> <p data-bbox="931 385 1469 469">For more information, see "Configuring Distributors in a Mixed eDirectory Environment" on page 328.</p>
Installing Inter-Server Communications	
13) Subscriber servers outside your secured network:	<p data-bbox="931 499 1469 560">If you do not need to set up inter-server communications, skip to worksheet item 14.</p> <p data-bbox="931 586 1469 701">Inter-server communications security might be needed if your Distributor and Subscriber servers communicate with servers outside your secured network.</p> <p data-bbox="931 727 1469 812">For more information, see "Determining Whether You Need Inter-Server Communications Security" on page 332.</p>
Installing NCI on Windows Servers	
14) Windows, Linux, or Solaris servers (Distributor or Subscriber) that will be involved with Distribution encryption:	<p data-bbox="931 1286 1469 1346">If you do not need to install NCI to Windows servers, skip to worksheet item 15.</p> <p data-bbox="931 1372 1469 1457">List the Windows, Linux, or Solaris servers that will either build (Distributors) or extract (Subscribers) encrypted Distributions.</p> <p data-bbox="931 1483 1469 1568">For more information, see "Determining Whether You Need Encryption Security for Windows Servers" on page 332.</p>

Configuration Information	Instructions
Configuring the Distributor Routing Hierarchies	
15) Distributors' routing hierarchies of tiered Subscribers:	<p>Create a chart of tiered Subscribers for each Distributor that shows how you want your Distributions to be distributed on your network. Distributors can use Subscribers in other Distributor's routing hierarchies. However, a Subscriber should only be used once in a given Distributor's hierarchy so that an end-node Subscriber will only have one distribution path for receiving a particular Distribution.</p> <p>For more information, see "Determining the Distribution Flow" on page 329.</p>
Configuring Parent Subscribers	
16) Subscriber/parent Subscriber assignments (end-node Subscribers associated with a parent Subscriber):	<p>Create Subscriber lists where each parent Subscriber will deliver Distributions. Each end-node Subscriber should be assigned to a parent Subscriber, except where you want the end-node Subscriber to receive its Distribution directly from the Distributor.</p> <p>For more information, see "Selecting Subscribers for the Distribution Routes" on page 330.</p>

Configuration Information	Instructions
Creating and Configuring Subscriber Groups	
17) Subscriber Group object name:	If you are not using Subscriber Groups, skip to worksheet item 19 . Enter a unique name for the Subscriber Group. For more information, see “Subscriber Groups” on page 334 .
18) Subscribers to be in this group:	Enter a list of Subscribers that need the same Distributions from the Channel where the group will be subscribed.
Creating the Policy Package Distributions	
19) Distributions, their types, and their Distributors:	Create a list of your Distributions. For each Distribution, include the Distribution type, object name, and servers that will need the Distribution. The Distribution types are: <ul style="list-style-type: none">◆ “File” on page 321◆ “FTP” on page 321◆ “HTTP” on page 321◆ “RPM” on page 322◆ “Desktop Application” on page 322◆ “Policy Package” on page 323◆ “Software Package” on page 322 For more information, see “Selecting Your Distributions” on page 320 .

Configuration Information	Instructions
Creating and Configuring the Channels	
20) eDirectory container for TED objects:	<p>Container for creating and managing TED objects.</p> <p>You might have created a TED container during installation of Policy and Distribution Services. If not, you should create a container specifically for managing TED objects.</p> <p>For more information, see “Determining Whether a Distributor Server Will Host a ZENworks Database” on page 327.</p>
21) Channel names:	<p>Enter the names of the Channel objects that you will need for your Distributions. We recommend a unique Channel for each unique Distribution or Distribution grouping.</p> <p>For more information, see “Determining the Channels for the Distributions” on page 333.</p>
22) Distributions for the Channels:	<p>Create a list of which Distributions belong to which Channels.</p> <p>For more information, see “Determining the Channels for the Distributions” on page 333.</p>

Subscribing to the Channels

23) Subscribers' Extract schedules:

Set extract schedules per Subscriber server according to when it would be best for each Subscriber to be extracting its Distributions.

For more information, see ["Determining Subscribers' Subscriptions"](#) on page 334.

24) Channel associations with Subscribers and Subscriber Groups:

Create lists where Subscribers and Subscriber Groups are associated with the Channels that have the Distributions you want them to receive.

15 Novell iManager

If you have not yet installed and set up the ZfS Management role in Novell® iManager, see [“Installing Web-Based Management for Policy and Distribution Services”](#) in the *Installation* guide.

The ZfS Management role in iManager enables you to manage Tiered Electronic Distribution (TED) objects, agents, and processes from any location where Internet Explorer 5.5 or later is available. Using the ZfS Management role, you can:

- ◆ Create, modify, and delete TED objects (Distribution, Subscriber, Distributor, Channel, Subscriber Group, and External Subscriber).
- ◆ View a graphical representation of your distribution system, which makes it easy to track a Distribution from Distributor to end node Subscriber, no matter how many parent Subscribers the Distribution passes through.
- ◆ Display a browser-based console, called the Remote Web Console, for each Distributor Agent, Subscriber Agent, and Policy/Package Agent in your system. From the Remote Web Console, you can check the configuration of any agent, monitor the activities of any agent, and control many agent functions.

The following sections help you make the most of the features available to you in the ZfS Management role:

- ◆ [“Accessing the ZfS Management Role in iManager” on page 359](#)
- ◆ [“Managing Tiered Electronic Distribution” on page 361](#)
- ◆ [“Monitoring the Distribution Process” on page 363](#)
- ◆ [“Monitoring Specific Agents” on page 364](#)
- ◆ [“Comparing the ZfS Management Role in iManager with ConsoleOne Capabilities” on page 368](#)

Accessing the ZfS Management Role in iManager

To access iManager in Internet Explorer:

- 1 Access the following URL:

```
http://server:port/eMFrame/iManager.html
```

where *server* is the IP address or DNS host name of the server where iManager is installed and *port* is either 8080 (the Tomcat default) or another port number set up when Tomcat was installed and configured.



If the iManager login page does not appear, check how you typed the URL. Make sure that you typed `eMFrame` and `iManager.html` exactly as shown in the example; they are case sensitive.

2 Log in to the Novell eDirectory™ tree where TED objects are located.

TIP: If you are running iManager on a Windows server where the Novell Client™ is not installed, specify the IP address of a server where a replica of your eDirectory tree resides, instead of the tree name itself.

If you cannot log in, contact the administrator who set up the Zfs Management role in iManager. You must be assigned to the ZFS Management role before you can log in to iManager to act in that role.

After you successfully log in, the main iManager page is displayed. The top frame provides buttons for features.



3 Move the mouse pointer over the buttons to familiarize yourself with their functions.

The mouse-over text appears to the right of the row of buttons.

4 Click Roles and Tasks.

5 In the left panel, expand ZFS Management to list the available tasks:



6 Continue with the task that you want to perform:

- ◆ “Managing Tiered Electronic Distribution” on page 361
- ◆ “Monitoring the Distribution Process” on page 363
- ◆ “Monitoring Specific Agents” on page 364

If an error message displays while you are using the Zfs Management role in iManager, see “Novell iManager Errors” in “Policy and Distribution Services” in the *Troubleshooting* guide.

Managing Tiered Electronic Distribution

Acting in the ZfS Management role in iManager, you can create, edit, and delete the following TED objects in eDirectory:

- Distributor
- Distribution
- Channel
- Subscriber
- Subscriber Group
- External Subscriber

For these TED objects, you can perform all of the same management tasks in iManager that you can perform in ConsoleOne®:

- ◆ [“Creating TED Objects in iManager” on page 361](#)
- ◆ [“Editing TED Object Properties in iManager” on page 362](#)
- ◆ [“Deleting TED Objects in iManager” on page 362](#)

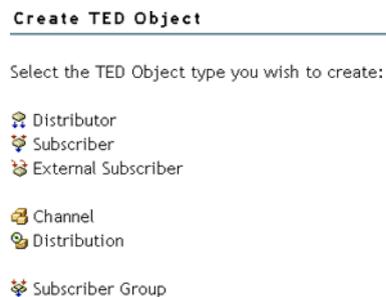
The following Policy and Distribution Services management tasks cannot be performed in iManager and must be performed using ConsoleOne:

- ◆ Creating, editing, and deleting policy packages. See [Chapter 17, “Server Policies,” on page 455](#).
- ◆ Creating, editing, and deleting software packages. See [Chapter 18, “Server Software Packages,” on page 493](#).
- ◆ Creating, editing, and deleting desktop applications. See [Chapter 19, “Desktop Application Distribution,” on page 525](#).
- ◆ Managing the Policy/Distribution database. See [Chapter 23, “ZENworks Database,” on page 587](#)
- ◆ Generating reports from the Policy/Distribution database. See [Chapter 24, “Reporting,” on page 599](#)

Creating TED Objects in iManager

To create a new TED object using iManager:

- 1** Click Roles and Tasks in the top frame > expand ZFS Management in the left frame > click Create TED Object.



- 2** Click the type of object you want to create.

- 3 Provide the information required for that object type, such as a unique name for the object, the context where you want to create the object, and so on.

Click Help for more information.

- 4 Click OK.
- 5 Continue with “[Editing TED Object Properties in iManager](#)” on page 362 to configure the new TED object.

Editing TED Object Properties in iManager

To edit the properties of a TED object using **iManager**:

- 1 Click Roles and Tasks in the top frame > expand ZFS Management in the left frame > click Edit TED Object.

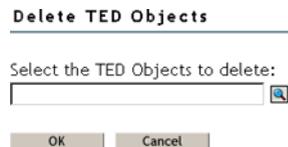


- 2 Browse to and click the TED object whose properties you want to edit > click OK.
The same property pages and options are available in iManager that are available in ConsoleOne.
You can click Help on each property page for information on setting the options.
- 3 Configure the object as needed > click OK to save the new properties settings.

Deleting TED Objects in iManager

To delete TED objects using **iManager**:

- 1 Click Roles and Tasks in the top frame > expand ZFS Management in the left frame > click Delete TED Object.



- 2 Browse to and click one or more TED objects to delete > click OK to list the objects on the Delete TED Objects page.
- 3 Click Help for information about the repercussions of deleting specific types of objects from your distribution system.
- 4 Click OK to delete the listed objects > click OK again to confirm.
- 5 Follow any instructions in the online help to reconfigure remaining objects so that the deletion does not disrupt your distribution system.

Monitoring the Distribution Process

The Tiered Distribution View enables you to track a Distribution from its Distributor through any parent Subscribers down to the end node Subscriber. This helps you determine which Subscribers have received the Distribution, where they received it from, and when they received it. This, in turn, helps you troubleshoot and correct any problems that may occur during the distribution process. This capability is not available in ConsoleOne.

To access the Tiered Distribution View in **iManager**:

- 1** Click Roles and Tasks in the top frame > expand ZFS Management in the left frame > click Tiered Distribution View.
- 2** Browse to and select the Distribution you want to track > click Next.
- 3** Select the Channel through which you want to track the Distribution > click Next.

The Distribution System window lists Subscribers that should receive the Distribution.

- 4** Click Expand All to display the routing hierarchy between the Distributor that built and sent the Distribution and the end node Subscribers that should have received it.

or

Click an individual server to expand its part of the hierarchy.

Icons indicate the status of the Distribution:

Icon	Meaning
	The Distribution has been received and extracted successfully.
	The Distribution has been received but not yet extracted. Check the Subscriber's extract schedule to see whether extraction has been attempted. If extraction was attempted and failed, check the Subscriber's event log to see what error occurred during extraction. See "Managing the TED Agents from the Remote Web Console" on page 364 .
	The Distribution was not successfully received by the Subscriber. Check the Subscriber's event log for an error message describing the problem. See "Managing the TED Agents from the Remote Web Console" on page 364 .
	The Distributor has not received any response from the Subscriber concerning the status of the Distribution. Check the status of the Subscriber and any parent Subscribers between it and the Distributor. See "Managing the TED Agents from the Remote Web Console" on page 364 .
	The Subscriber is running ZfS 2 software, rather than ZfS 3.0.2 software. Therefore, no status information is available in the Tiered Distribution View.

- 5** To display status information, select a Distributor or Subscriber > click Remote Web Console.
For information about the types of status information you can obtain, see ["Monitoring Specific Agents" on page 364](#).
- 6** To check configuration information, select a Distributor or Subscriber > click eDirectory Configuration.

You can edit the Distributor or Subscriber object properties just as if you had clicked Edit TED Object under ZFS Management. The same property pages and options are available in iManager that are available in ConsoleOne.

Monitoring Specific Agents

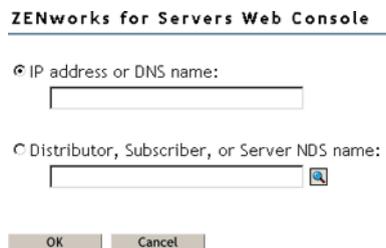
On NetWare[®] servers, you can monitor the TED agents (Distributor Agent and Subscriber Agent) and the Policy/Package Agent at the server console where they are running. In addition, you can monitor the agents running on any platform (NetWare, Windows, or UNIX) from Internet Explorer using the Zfs Management role in iManager.

- ◆ “Managing the TED Agents from the Remote Web Console” on page 364
- ◆ “Managing the Policy/Package Agent from the Remote Web Console” on page 366
- ◆ “Opening Multiple Remote Web Console Windows” on page 368

Managing the TED Agents from the Remote Web Console

To access the Remote Web Console for a Distributor or Subscriber in iManager:

- 1 Click Roles and Tasks in the top frame > expand ZFS Management in the left frame > click Remote Web Console.



- 2 Specify the IP address or DNS host name of a server where the Distributor Agent or Subscriber Agent is running.

or

Browse to and select a Distributor or Subscriber object or the Server object representing the server where the Distributor Agent or Subscriber Agent is running.

- 3 Click OK.
- 4 In the Available Services field, select Tiered Electronic Distribution.

Tabs at the top of the Remote Web Console frame provide various types of information related to the TED agents. Additional options are available on each tab.



You can click Help on each Remote Web Console page for information on using the features available on that page.

- 5 Continue with the task that you want to perform:
 - ◆ “Managing TED Objects” on page 365
 - ◆ “Monitoring TED Agent Status” on page 365

- ◆ [“Monitoring Distribution Status” on page 366](#)
- ◆ [“Forcing TED Agent Actions” on page 366](#)
- ◆ [“Managing Security Certificates” on page 366](#)

The tables below summarize these tasks, detail the Remote Web Console tab and option to use for each task, and indicate whether the task can also be performed using ConsoleOne.

Managing TED Objects

TED Agent (Distributor and Subscriber) Management Task	Remote Web Console Tab and Option	ConsoleOne
List all object properties of Distributor and Subscriber objects in a single list	Configuration > Configuration	No
List the object properties of any subordinate Subscriber in the routing hierarchy	Configuration > Subordinate Configuration	No
List all object properties of Distribution objects (except type-specific information) in a single list	Distributions > Distribution Information	No
List all object properties of Channel objects in a single list	Channels > Channel Information	No
Display information about the Policy/Distribution database	Configuration > Database	Yes

Be aware that if the Distributor has not been refreshed since changes were made to object properties in eDirectory, the object properties displayed in the Remote Web Console will be different from the object properties displayed in ConsoleOne. The Remote Web Console displays object information from the point of view of the Distributor Agent.

Monitoring TED Agent Status

TED Agent (Distributor and Subscriber) Management Task	Remote Web Console Tab and Option	ConsoleOne
View and continuously refresh the current Distributor event log, complete with message severity levels	Events > Distributor Event Log	No
View and continuously refresh the current Subscriber event log, complete with message severity levels	Events > Subscriber Event Log	No
Display the current status of the various distribution threads started by the TED agents to perform their various functions	Configuration > Threads	No

You can look up error messages that appear in the event logs in [“Tiered Electronic Distribution Errors”](#) in [“Policy and Distribution Services”](#) in the *Troubleshooting* guide.

Monitoring Distribution Status

TED Agent (Distributor and Subscriber) Management Task	Remote Web Console Tab and Option	ConsoleOne
List all Distributions currently being processed by the Distributor and/or Subscriber, along with detailed status information	Distributions > Active Distributions	No
Display status information for a selected Distribution that has been received by a Subscriber	Distributions > Received Distributions	No
Display the route that a Distribution must take through the routing hierarchy from a Distributor or parent Subscriber to any subordinate Subscriber	Configuration > Route to Subscriber	No

Forcing TED Agent Actions

TED Agent (Distributor and Subscriber) Management Task	Remote Web Console Tab and Option	ConsoleOne
Immediately refresh a Distributor so that it rereads eDirectory to check for modified Distributions	Configuration > Refresh Distributor	Yes
Immediately build a Distribution	Distributions > Build Distribution	Schedule dependent
Immediately send to Subscribers all Distributions listed in a selected Channel	Channels > Distribute Channel	Not with one click

Managing Security Certificates

TED Agent (Distributor and Subscriber) Management Task	Remote Web Console Tab and Option	ConsoleOne
List the security certificates that are available on a Subscriber	Security > Show Certificates	No
Delete security certificates from a Subscriber	Security > Show Certificates > Delete	No
Have the Distributor sign Subscribers' Certificate Signing Request (.CSR) files so that the Subscribers can receive encrypted Distributions from the Distributor	Security > Sign CSR	Yes

Managing the Policy/Package Agent from the Remote Web Console

The Policy/Package Agent always runs along with the Subscriber Agent. It is responsible for installing the software and enforcing the policies that the Subscriber Agent receives and extracts. The Remote Web Console enables you to manage the Policy/Package Agent, which is not possible using ConsoleOne.

To access the Remote Web Console for a Policy/Package Agent in **iManager**:

- 1 Click Roles and Tasks in the top frame > expand ZFS Management in the left frame > click Remote Web Console.

ZENworks for Servers Web Console

☉ IP address or DNS name:

☉ Distributor, Subscriber, or Server NDS name:
 

- 2 Specify the IP address or DNS host name of a server where the Subscriber Agent is running.
or

Browse to and select a Subscriber object or the Server object representing the server where the Subscriber Agent is running.

- 3 Click OK.

- 4 In the Available Services field, select Policy/Package Agent.

Tabs at the top of the Remote Web Console frame provide various types of information related to the Policy/Package Agent.



You can click Help on each Remote Web Console page for information on using the features available on that page.

- 5 Continue with the task that you want to perform.

The table below summarizes these tasks and details the Remote Web Console tab for each task.

Policy/Package Agent Management Task	Remote Web Console Tab	ConsoleOne
List the plug-ins that are currently loaded for enforcing server policies	Configuration	No
List all the variables that the Policy/Package Agent has values for	Configuration	No
List all the policies that the Policy/Package Agent enforces on a Subscriber server	Policies	No
Immediately enforce one or more policies on a Subscriber server	Policies	No
Remove individual policies from a Subscriber server	Policies	No
Immediately refresh one or more policies so that the Distributor Agent rereads eDirectory to check for modifications	Policies	No
List all the software packages that the Policy/Package Agent installs on the Subscriber server	Software Packages	No
Determine the current status of all software packages installed on the Subscriber server	Software Packages	No
Create and run a program or script on the Subscriber server once or repeatedly	Schedule	No

Policy/Package Agent Management Task	Remote Web Console Tab	ConsoleOne
Down the Subscriber server	Actions	No
Restart the Policy/Package Agent (independent from the Subscriber Agent)	Actions	No

Opening Multiple Remote Web Console Windows

On any Remote Web Console page, click Detach in the upper right corner to display the current page in a new browser window. This enables you to access multiple Remote Web Console features at the same time. For example, you could detach one window for the TED agents and another window for the Policy/Package Agent. Or you could detach a window for the Remote Web Console and still be able to perform other ZfS Management tasks in the main Novel iManager window.

Comparing the ZfS Management Role in iManager with ConsoleOne Capabilities

The following table summarizes the major similarities and differences between the ZfS Management role in iManager and the capabilities provided in ConsoleOne:

Task	ZfS Management Role in iManager	ConsoleOne
Creating, editing, and deleting the following TED objects: Distributor Subscriber Distribution Channel Subscriber Group External Subscriber	Yes	Yes
Creating, editing, and deleting the following Policy and Distribution Services components: Policy Package Server Software Package Desktop Application	No	Yes
Setting up the following Distribution types: Policy Package Software Package Desktop Application HTTP FTP File RPM	Yes	Yes
Immediately refreshing a Distributor	Yes	Yes
Immediately building a Distribution	Yes	Not with one click

Task	ZfS Management Role in iManager	ConsoleOne
Immediately sending to Subscribers all Distributions listed in a Channel	Yes	Not with one click
Monitoring TED agent event logs and status	Yes	No
Listing and managing the policies on a Subscriber server	Yes	No
Listing and checking the status of software packages installed on a Subscriber server	Yes	No
Running programs and scripts on a Subscriber server	Yes	No
Downing a Subscriber server	Yes	No
Managing security certificates:		
Listing available certificates	Yes	No
Resolving certificates	No	Yes
Signing CSRs	Yes	Yes
Managing the Policy/Package Agent	Yes	No

16 Tiered Electronic Distribution

Novell® ZENworks® for Servers (ZfS) provides Tiered Electronic Distribution (TED) for managing distributions of files, policies, and software across your network.

TED is integrated with other Novell network management applications that snap in to the ConsoleOne® framework to take advantage of Novell eDirectory™ management and file access control. TED can also be managed using the ZfS Management role in Novell iManager.

For information on TED, see the following sections:

- ◆ “Understanding Tiered Electronic Distribution” on page 371
- ◆ “Common Distribution Tasks” on page 377 (a mini-index)
- ◆ “Distributors” on page 379
- ◆ “Distributions” on page 393
- ◆ “Channels” on page 415
- ◆ “Subscribers” on page 418
- ◆ “Subscriber Groups” on page 424
- ◆ “External Subscribers” on page 426
- ◆ “Configuring Multiple TED Objects” on page 433
- ◆ “Sending Distributions” on page 439
- ◆ “TED Issues” on page 442
- ◆ “Working Directories” on page 447
- ◆ “Editing the TEDNODE.PROPERTIES File” on page 451

Understanding Tiered Electronic Distribution

Review the following sections for an understanding of Tiered Electronic Distribution (TED):

- ◆ “Distribution Management through Tiered Electronic Distribution” on page 372
- ◆ “The Basic Distribution Process” on page 372
- ◆ “TED’s eDirectory Objects” on page 373
- ◆ “Physical Network Connections” on page 374
- ◆ “Distribution Flow Details” on page 374
- ◆ “Relationships of the TED Objects” on page 373
- ◆ “The ZfS Agents Used by TED” on page 374
- ◆ “The Tiered Distribution Model” on page 376

- ◆ [“TED’s Key Components” on page 377](#)

Distribution Management through Tiered Electronic Distribution

Tiered Electronic Distribution (TED) provides you with a way to manage your servers through the distribution of electronic data between servers. For example, application programs, collections of data files, software patches, and server policies.

When you install Policy and Distribution Services, the installation process creates TED and server policy objects in the eDirectory tree, copies software to the various servers, and sets up basic configurations for the TED and Server Policies components according to your installation selections.

The TED software can be hosted on NetWare[®], Windows NT, Windows 2000, Linux, and Solaris servers.

TED uses a tiered distribution model that enables one server to indirectly service hundreds or even thousands of other servers. TED makes it easy to distribute files and policy packages by building them into compressed data files and hosting them in distribution channels for dissemination to the appropriate servers.

TED lets you schedule the distribution processes to take advantage of off-peak hours. It also sends notification of distribution status by sending e-mail messages, logging events, displaying real-time messages, database reporting, and sending SNMP traps.

The Basic Distribution Process

The TED distribution process is based on the creation of Distributions (compressed file collections) that you use to move files and policies to your network servers. For more information, see [“Understanding the Distribution Processes” on page 440](#).

Following is a simplified distribution process. It is governed by **schedules** that you set for each of the TED objects involved with the Distribution file.

1. A Distributor creates a **security certificate** to provide distribution security.
2. A Distribution is built on the Distributor server’s file system according to the configuration you create in the **Distribution object**.
3. You associate the Distribution with a **Channel**.
4. You **subscribe** your target **Subscriber servers** to the Channel. This will cause them to receive all of the Distributions contained in that Channel.
5. The certificate (from 1 above) is copied to Subscriber servers for Distribution security verification.
6. The Channel’s listed Distributions are sent from the Distributor to the Subscriber servers whose security certificates are valid.
7. The Subscriber extracts the files or policies from the compressed Distribution file and applies them according to the Distribution object’s configuration.

The schedules that you need to coordinate for sending Distributions are the Distributor’s Refresh schedule, the Distribution’s Build schedule, and the Channel’s Send schedule.

The schedules that you need to coordinate for receiving and extracting Distributions are the Channel’s Send schedule and the Subscriber’s Extract schedule.

For information on scheduling, see “[TED Object Scheduling Issues](#)” on page 568.

TED’s eDirectory Objects

TED uses eDirectory objects and the related software for performing its distribution functions. The Distinguished Name (DN) of all TED objects includes the server name and component function of the host server.

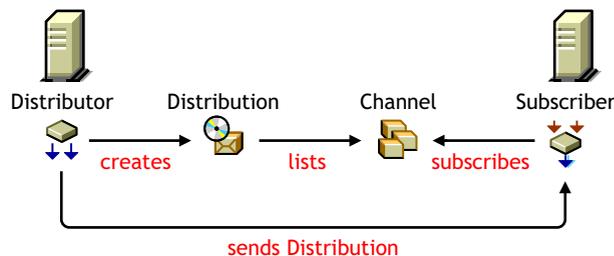
The eDirectory schema extensions included in TED define the classes of eDirectory objects that can be created in your eDirectory tree, including information that is required or optional at the time the object is created. Every object associated with TED in an eDirectory tree has a class defined for it in the tree’s schema.

You will extend the schema of your tree for the following eDirectory objects when you install ZfS 3.0.2:

TED Object	Basic Function	More Information
Distributor	Build, send Distributions	“ Distributors ” on page 379
Distribution	Contain files, policies	“ Distributions ” on page 393
Channel	List Distributions	“ Channels ” on page 415
Subscriber	Receive, extract Distributions	“ Subscribers ” on page 418
Subscriber Group	Channel subscriptions by multiple Subscribers	“ Subscriber Groups ” on page 424
External Subscriber	Enable distributing between trees	“ External Subscribers ” on page 426

Relationships of the TED Objects

The following illustrates the relationships of the main TED objects:



Note the following from this illustration:

- ◆ A Distributor creates a Distribution
- ◆ The Distribution is listed in a Channel
- ◆ A Subscriber subscribes to the Channel
- ◆ The Subscriber receives the Distribution from the Distributor (possibly via a parent Subscriber)

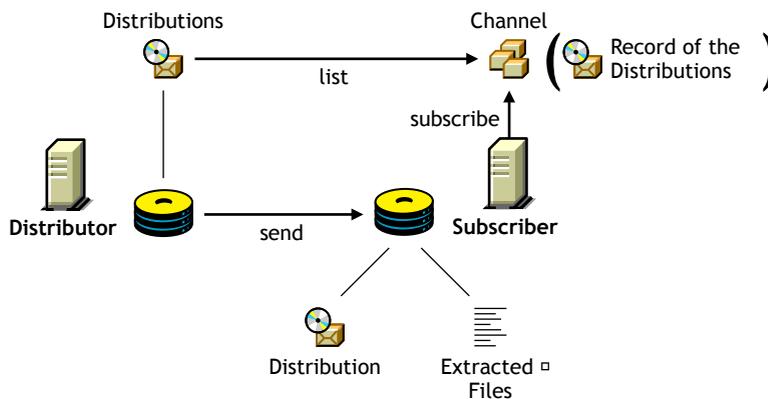
Physical Network Connections

Distributor and Subscriber servers can be physically connected to the network in any configuration, including having some servers across WAN links. The following describes the possible physical interactions between Distributor and Subscriber servers:

- ◆ A Subscriber server can be in the same geographic location as its Distributor server
- ◆ A Subscriber server can be in a different geographic location from its Distributor server, such as across a WAN link
- ◆ A Distributor server can service multiple Subscriber servers
- ◆ A Subscriber server can be serviced by multiple Distributor servers
- ◆ A Subscriber server can receive its Distribution files directly from a Distributor server
- ◆ A Subscriber server can receive its Distribution files indirectly via another Subscriber server acting as a parent Subscriber

Distribution Flow Details

The following illustrates the physical flow of TED Distributions:



Note the following from the illustration:

- ◆ A Distribution file is stored on the Distributor server's hard drive
- ◆ The Channel lists a Distribution (it does not hold a copy of the Distribution)
- ◆ The Subscriber subscribes to a Channel to obtain all of the Distributions listed there
- ◆ The Subscriber extracts the Distribution contents from the file's compressed format and writes the content to the volume and directory specified in the Distribution's configuration

IMPORTANT: When there are multiple versions of a File or Desktop Application Distribution, the Subscriber maintains copies of each of the versions, as is specified in the Distribution object's properties. The default is to maintain 10 versions per Distribution type.

The ZfS Agents Used by TED

The following ZfS agents are used to perform the TED distribution process' actual functions:

- ◆ [“Distributor Agent” on page 375](#)
- ◆ [“Subscriber Agent” on page 375](#)

- ◆ “Policy/Package Agent” on page 375

Distributor Agent

The Distributor Agent is installed on each server where you select the Distributor option on the Server Selection for Policy and Distribution Services installation page.

This TED agent has the following functions:

- ◆ Builds Distributions based on the information contained in the Distribution objects that are associated with the Distributor.
- ◆ Reads eDirectory for all TED configuration information (Distribution, Channel, and Subscriber information), and builds and sends Distributions accordingly.
- ◆ Handles all notifications and events for the Subscriber.
- ◆ Sends DS configuration information found in Subscriber objects to each Subscriber as part of each Distribution.
- ◆ Adheres to Distribution schedules for building the Distributions belonging to a Distributor.
- ◆ Adheres to Channel schedules for sending Subscribers’ configuration information and any of the Distributions listed in the Channel.
- ◆ Logs Tiered Electronic Distribution information to the TED.LOG file for reporting purposes

Subscriber Agent

The Subscriber Agent is installed on each server where you select the Subscriber/Policies option on the Server Selection for Policy and Distribution Services installation page.

This TED agent has the following functions:

- ◆ Subscribes its Subscriber server to Channels for receiving Distributions.
- ◆ Receives and extracts (installs) the following Distribution types to the server’s file system:
 - File
 - FTP
 - HTTP
 - RPM
 - Desktop Application ¹

¹ The Desktop Application Distribution is only available when ZENworks for Desktops (ZfD) is installed.

- ◆ Receives and hands off some Distribution types to the Policy/Package Agent (see “Policy/Package Agent” on page 460).
- ◆ In the parent Subscriber role, receives a Distribution and forwards it on to other Subscriber servers.
- ◆ Logs Tiered Electronic Distribution information to the TED.LOG file for reporting purposes

Policy/Package Agent

The Policy/Package Agent is installed on each server where you select the Subscriber/Policies option on the Server Selection for Policy and Distribution Services installation page.

This Server Policies agent has the following TED functions:

- ◆ Extracts and enforces policy information from Policy Package Distributions.
- ◆ Extracts and installs the contents of Server Software Packages (Software Package Distributions).
- ◆ Logs policy and software package information to the DEFAULTLOG.TXT file for reporting purposes

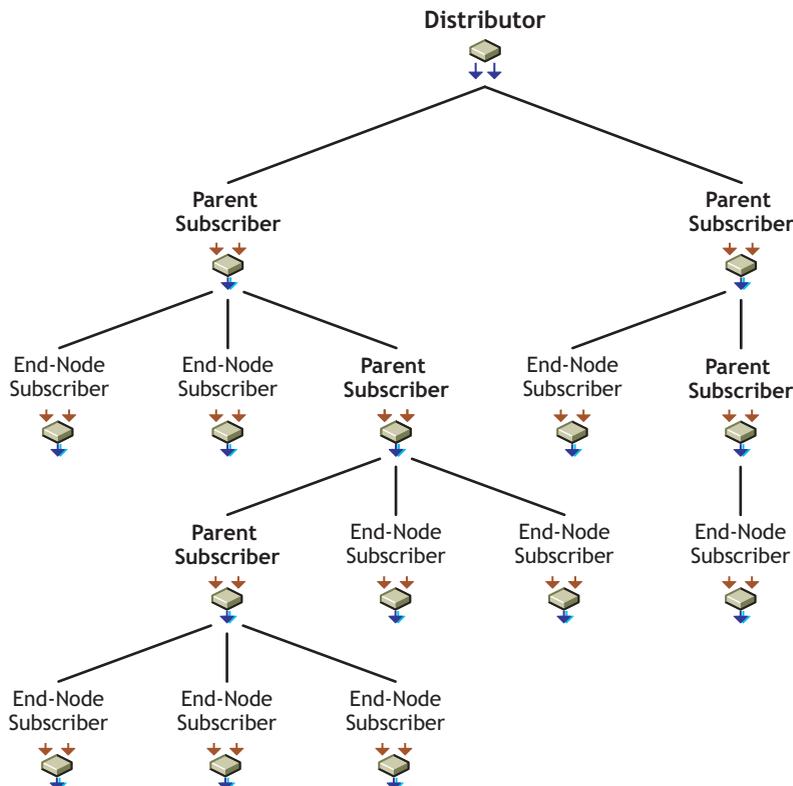
This agent has other policy-related functions. For more information, see [Chapter 17, “Server Policies,”](#) on page 455.

The Tiered Distribution Model

The power of the tiered distribution model is that you can spread the workload for sending Distributions. This is particularly important to the Distributor servers. By sharing distribution duties with parent Subscribers, a Distributor server can have more resources available for reading eDirectory, building each of its Distributions, and logging information to the database.

Tiered distribution levels can be very deep, providing a very large number of Subscribers that any one Distributor can service—without doing so directly.

The following illustrates a distribution routing hierarchy containing a Distributor, several parent Subscribers, and many end-node Subscribers:



The Distributor can service hundreds of parent Subscribers directly, or service just a few first-tier parent Subscribers and let them do the bulk of the distribution work. In the above illustration, the Distributor only has to send its Distribution to two parent Subscribers, yet nine end-node Subscribers will receive the Distribution.

The parent Subscribers shown in this illustration can also receive the Distribution for extraction if they were also subscribed to the Distribution's Channel. If all of the parent Subscribers in the above illustration were subscribed to receive the Distribution being sent to the end-node Subscribers, the Distributor will have serviced 14 total Subscriber servers while only itself sending the Distribution twice.

Each parent Subscriber can service hundreds of other parent Subscribers or end-node Subscribers (the intended recipients of the Distributions). The workload for passing on a Distribution by a parent Subscriber is minimal in compared to the workload for the Distributor to build the Distribution.

As you can see, the tiered distribution model allows you to minimize the distribution workload for your Distributor servers.

TED's Key Components

In summary, the key components of TED include:

- ◆ eDirectory schema extensions that include objects for Distributors, Distributions, Channels, Subscribers, and External Subscribers
- ◆ ConsoleOne snap-ins and iManager plug-ins that provide creation, configuration, and management of TED
- ◆ A Distributor Java process hosted on a NetWare, Windows NT, Windows 2000, Linux, or Solaris server for handling distribution of data packages to Subscribers
- ◆ A Subscriber Java process hosted on a NetWare, Windows NT, Windows 2000, Linux, or Solaris server that subscribes to a Channel for its Distributions
- ◆ A routing hierarchy for each Distributor that has a hierarchical list of Subscribers who can both receive Distributions for themselves and pass the Distributions on to other Subscribers
- ◆ Parent Subscribers that pass Distributions on to other Subscribers
- ◆ An External Subscriber object that allows distributing between trees or to servers that do not have eDirectory server objects
- ◆ The Distributor Agent that controls the actual processes of building the Distribution files on the Distributor, and the Subscriber Agent that controls extracting from Distribution files on the Subscriber
- ◆ Policy/Package Agent that extracts and enforces policy information from Policy Package Distributions, and extracts and installs the contents of software packages
- ◆ Certificates that provide distribution security

Common Distribution Tasks

The following tables provide documentation links to common TED tasks. All links are to sections in this Policy and Distribution Services portion of the *Administration* guide.

TED Object Tasks	Instructions
Create a Distributor or Subscriber	"Reinstalling ZENworks for Servers" under "Installing ZENworks for Servers" in the <i>Installation</i> guide.

TED Object Tasks	Instructions
Configure multiple TED objects	“Configuring Multiple TED Objects” on page 433
Change the DNS name or IP address of a TED server	“Changing DNS Names or IP Addresses for TED Servers” on page 446
<hr/>	
Distributor Tasks	Instructions
Configure a Distributor object	“Configuring Distributors” on page 389
Create a routing hierarchy for a Distributor	“Understanding Distribution Routing” on page 381 and “Configuring Distributors” on page 389
Delete a Distributor object	“Deleting a Distributor Object and How Its Distributions Are Affected” on page 392
Refresh a Distributor	“Refreshing the Distributor” on page 392
Create a security certificate on a Distributor and copy it to its associated Subscribers	“Creating Security Certificates for Non-Encrypted Distributions” on page 557
<hr/>	
Distribution Tasks	Instructions
Create a Distribution	“Distributions” on page 393
Delete a Distribution	“Deleting a Distribution” on page 410
Managing orphaned Distributions (when their Distributor object has been deleted)	“Deleting a Distributor Object and How Its Distributions Are Affected” on page 392
Schedule and send a Distribution	“Sending Distributions” on page 439
Force a Distribution to be sent	“Forcing a Single Distribution To Be Sent” on page 441
Use a parent Subscriber to send a Distribution	“Sending Distributions Through Parent Subscribers” on page 441
Send a Distribution to another tree	“Sending Distributions Between Trees” on page 442
Import or export a Distribution manually	“Manually Importing and Exporting Distributions” on page 412
Create and send a File Distribution using a wizard	“Using the TED Distribution Wizard” on page 414
<hr/>	
Channel Tasks	Instructions
Create a Channel	“Creating and Configuring Channels” on page 417

Channel Tasks	Instructions
Force a Channel to fire	“Forcing a Channel To Be Sent” on page 418

Subscriber Tasks	Instructions
Configure a Subscriber object	“Configuring Subscribers” on page 420
Create an External Subscriber object	“Creating and Configuring External Subscribers” on page 432
Configure the TEDNODE.PROPERTIES file for a Subscriber server that does not have its own configuration capability	“Editing the TEDNODE.PROPERTIES File” on page 451

Network Traffic Management Tasks	Instructions
Control bandwidth usage for Distribution traffic by setting the I/O rates	“Controlling I/O Rates and Concurrent Distributions” on page 444
Minimize network messaging traffic	“Minimizing Messaging Traffic” on page 445

Distributors

The following sections provide concepts and instructions for the Distributor object:

- ◆ [“Understanding Distributors” on page 379](#)
- ◆ [“Understanding Distribution Routing” on page 381](#)
- ◆ [“Creating Distributors” on page 389](#)
- ◆ [“Configuring Distributors” on page 389](#)
- ◆ [“Refreshing the Distributor” on page 392](#)
- ◆ [“Deleting a Distributor Object and How Its Distributions Are Affected” on page 392](#)

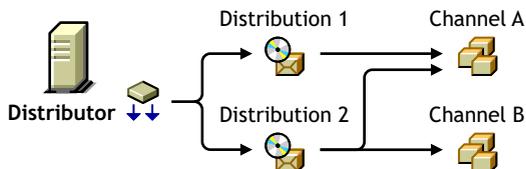
Understanding Distributors

The Distributor object (TED Distributor) is an eDirectory object that defines the properties for the Distributor.

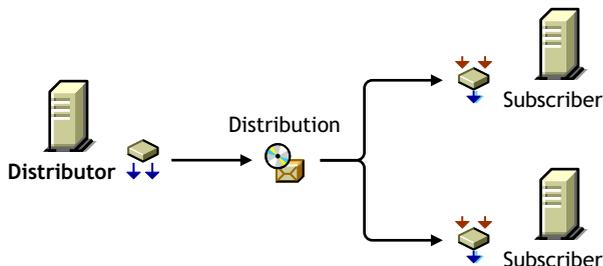
- ◆ [“Functional Relationship with Other TED Objects” on page 380](#)
- ◆ [“Distributor Description” on page 380](#)
- ◆ [“Scheduling” on page 380](#)
- ◆ [“Routing Distributions” on page 381](#)
- ◆ [“Multiple Distributors in the Tree” on page 381](#)
- ◆ [“Database Logging” on page 381](#)

Functional Relationship with Other TED Objects

The following illustrates that a Distributor can list any one of its Distributions in several Channels, and several of its Distributions in one Channel:



The following illustrates that a Distributor sends its Distributions to Subscriber servers:



Distributor Description

The Distributor server's main TED function is to create and send Distributions. It also logs information to a database file, if you have one assigned for the Distributor.

The Distributor Agent builds a Distribution file on the Distributor server from the information you provide when you create and configure a Distribution object. A Distributor can own multiple Distributions.

When a Distributor builds a Distribution, it can optionally create a digest that provides a checksum for the Subscriber to compare against. Digests are used by Subscribers to verify that the Distributions have not been tampered with while in transit. Creating a digest is optional per Distributor, so the digests might not always be available for a checksum comparison by any Subscriber where this option is enabled.

A Distributor lists its Distributions in Channels. Distributors do not own Channels. However, a Distributor is the sole owner of its Distributions.

The Distributor sends its Distributions to Subscribers (usually parent Subscribers for passing on the Distributions). If an end-node Subscriber does not respond to a Distributor (or a parent Subscriber) that is trying to send a Distribution to it, the Distributor will retry sending a Distribution every two minutes for 30 minutes, then stop. It will not attempt to re-send the Distribution until the Channel's Send schedule starts again.

Scheduling

A Distributor's Refresh schedule determines when it will read eDirectory for changes to its Distributions and other TED objects. A Distributor builds all new Distributions it finds and rebuilds any of its Distributions that have changed. The new or rebuilt Distributions are then available to be sent when a Channel's Send schedule starts.

IMPORTANT: We recommend the Distributor's Refresh schedule be daily, unless changes to Distributions warrant a more frequent refresh. However, do not refresh the Distributor more often than every five minutes.

The following can need up to five minutes to complete their processes: Distribution building, eDirectory replication, and tree walking (when no Search policy is defined).

A Distributor can build its Distributions any time its Refresh schedule starts, or you can force it to do so from the server's command line.

If you delete a Distribution, you should also refresh the Distributor immediately so that it will recognize the deletion and not try to build a Distribution that no longer exists. For information on deleting Distributions, see [“Deleting a Distribution” on page 410](#).

For information on scheduling, see [Chapter 21, “Scheduling,” on page 565](#).

Routing Distributions

The Distributor contains a distribution route, which is a hierarchical list of Subscribers that indicate the routes the Distributor can take to send its Distributions to its Subscriber servers. For information on routing hierarchies, see [“Understanding Distribution Routes” on page 329](#).

Multiple Distributors in the Tree

You can have multiple Distributor objects in the tree; however, you can only have one Distributor installed per server. The need for multiple Distributors is dependent on several factors. For more information, see [“Are Additional Distributors Needed?” on page 325](#).

Database Logging

Individual Distributors can log information to their own database files, or all Distributors can log information to one common database file. For information on databases, see [Chapter 23, “ZENworks Database,” on page 587](#).

Understanding Distribution Routing

A distribution route represents the most efficient path to any given segment of your WAN. A distribution route is a list of parent Subscribers that relay Distributions on to other parent or end-node Subscribers. Parent Subscribers can be used to minimize the workload for a Distributor because they can pass on Distributions to other Subscribers.

The following sections explain how a Distributor moves its Distributions to your network's servers:

- ◆ [“Understanding Parent Subscribers” on page 381](#)
- ◆ [“Understanding Routing Hierarchies” on page 383](#)
- ◆ [“Sharing Parent Subscribers with Other Distributors” on page 385](#)
- ◆ [“Distributing Across WAN Links” on page 387](#)
- ◆ [“Out-of-Tree Distributions” on page 388](#)
- ◆ [“Routing Hierarchy Configuration Guidelines” on page 388](#)

Understanding Parent Subscribers

A parent Subscriber is a Subscriber that acts as a proxy for the Distributor to store and pass Distributions so that the Distributor does not have to send its Distributions directly to every Subscriber. Parent Subscriber servers do not need to be recipients themselves of a Distribution to temporarily store it for passing on to other Subscriber servers.

- ◆ “Distributors Send Distributions Using Parent Subscribers” on page 382
- ◆ “Passing on Unsubscribed Distributions” on page 382
- ◆ “Sharing the Distribution Load” on page 382
- ◆ “Balancing Workloads” on page 382

Distributors Send Distributions Using Parent Subscribers

A Distributor server must actually send each of its Distributions, because the Distribution files reside in its own file system.

Sending Distributions can create an enormous workload for a Distributor if it has to individually send each of its Distributions to every Subscriber server on the network. Therefore, parent Subscribers are used to help send Distributions.

A detailed understanding of your network’s topology is important for properly configuring distribution routes and selecting parent Subscribers. If necessary, create a diagram of your network that shows all WAN links to determine how to use parent Subscribers.

Passing on Unsubscribed Distributions

A Subscriber does not have to subscribe to a Channel containing a Distributor’s Distributions to be in the Distributor’s routing hierarchy. A parent Subscriber itself does not need to be the recipient of the Distribution it is passing on.

Further, a parent Subscriber does not have to subscribe to the same Channels as its subordinate Subscribers to be able to pass on those Channel’s Distributions.

Sharing the Distribution Load

In the illustration under “The Routing Hierarchy” on page 384, each Subscriber listed could be a parent to other Subscribers on its LAN. For example, if every Subscriber listed in the illustration was a parent to 20 end-node Subscribers, the Distributor could service 210 total Subscribers while only physically sending its Distributions to three of the Subscribers (the first-tier parent Subscribers, numbers 01, 04, and 09).

To further illustrate, parent Subscriber 04 would be servicing 104 Subscribers while only directly sending to two parent Subscribers (05 and 06) and its own 20 end-node Subscribers.

Balancing Workloads

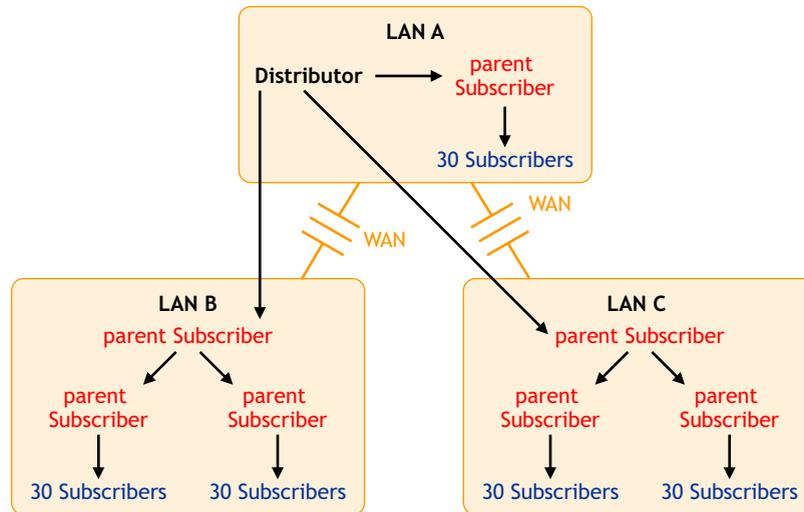
A Distributor can use parent Subscribers in a routing hierarchy to explicitly determine routes for its Distributions. This eases its workload in distributing to Subscribers.

A parent Subscriber can also help a Distributor with its workload by acting as a proxy for the Distributor to pass on Distributions to other Subscribers. You can have multiple parent Subscribers on a given LAN to share the distribution workload on the LAN.

We estimate that the number of Subscribers and/or parent Subscribers that any one Distributor or parent Subscriber should service to be about 40. This figure is dependent on such factors as network speed, sizes of Distributions, and so on.

You should place parent Subscribers where they will help in load-balancing for Distributors and other parent Subscribers.

The following illustrates a WAN environment with parent Subscribers:



Note the following from this illustration:

- ◆ Assume that the three parent Subscribers that the Distributor’s distribution lines point to are the first-tier Subscribers in the Distributor’s routing hierarchy.
- ◆ Assume that the other four parent Subscribers (in LAN B and LAN C) are listed in the second tier of the distribution hierarchy.
- ◆ The Distributor does not need to send the Distributions directly to the 30 Subscribers on LAN A because the parent Subscriber in LAN A will do that.
- ◆ The Distributor only sends its Distributions directly to the three parent Subscribers, but a total of 157 Subscribers can receive those Distributions.
- ◆ One parent Subscriber in LAN B (and the same for LAN C) was used solely for receiving Distributions directly from the Distributor, then passing them on to other parent Subscribers, which in turn passed them to their 60 Subscribers. For large systems, this scheme can make a parent Subscriber on the other side of a WAN link more available to a Distributor, instead of that parent Subscriber being so busy passing Distributions to its many other end-node Subscribers that it can make the Distributor wait. Consider this hierarchical design where it might be applicable in your network.

The Distributor has the workload of reading eDirectory for Distribution changes, building the Distributions, sending the Distributions, and writing to the ZENworks database. By minimizing the number of Subscribers that a Distributor itself must directly send Distributions to, you can give the Distributor more resources for its various functions.

Understanding Routing Hierarchies

TED provides a routing hierarchy to automate sending your Distributions from the Distributor servers to your Subscriber servers.

- ◆ [“The Routing Hierarchy” on page 384](#)
- ◆ [“Distributing Using the Hierarchy” on page 384](#)
- ◆ [“Subscribers Orphaned from the Routing Hierarchy” on page 385](#)
- ◆ [“Rerouting Because of Changes to the Routing Hierarchy” on page 385](#)

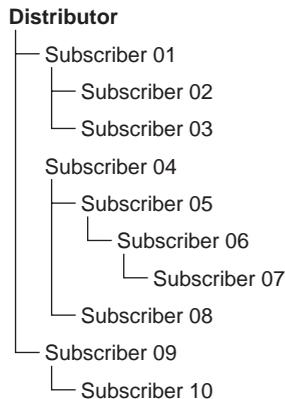
The Routing Hierarchy

To ease a Distributor's workload in sending Distributions, each Distributor has its own routing hierarchy, which is a hierarchical list of Subscribers that indicate the routes Distributions can take to send a Distribution to a Subscriber. The Subscribers in the routing hierarchy are the parent Subscribers. Parent Subscribers can be nested many levels deep.

A parent Subscriber can receive a Distribution and extract it, as well as pass that same Distribution on to other Subscribers.

You can modify distribution routes at any time by editing the properties of the Distributor objects.

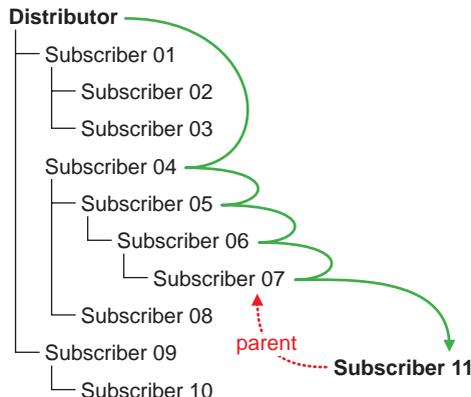
The following illustrates a Distributor's routing hierarchy:



Note that the only Subscribers you need to include in the Distributor's routing hierarchy are those that will be used to pass on Distributions to other Subscribers. Subscribers that are not used to pass on Distributions can be referred to as end-node Subscribers.

Distributing Using the Hierarchy

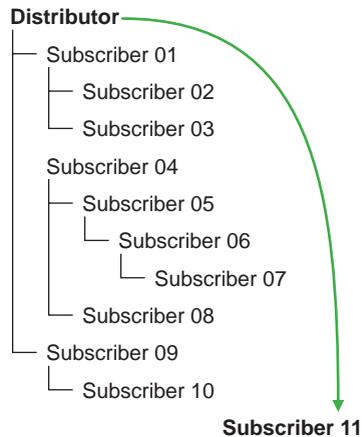
Assume that Subscriber 07 is a parent to Subscriber 11 (which is not in the routing hierarchy). The distribution route from the Distributor to Subscriber 11 would be the following:



The Distributor used four parent Subscribers (04, 05, 06, and 07) to send the Distribution to Subscriber 11.

Subscribers Orphaned from the Routing Hierarchy

If Subscriber 11 did not have a parent Subscriber (such as Subscriber 07), the Distribution would come directly from the Distributor:



Note that the only Subscribers you need to include in a routing hierarchy are those that will be used to pass Distributions on to other Subscribers. The end-node Subscribers (Subscribers that are only receiving and not passing on Distributions) do not need to be listed in the hierarchy. They have links in eDirectory to their parents.

Subscribers that exist in a routing hierarchy are generally parent Subscribers, although this is not required.

IMPORTANT: Subscribers that do not utilize parent Subscribers can increase the workload on the Distributor and increase network traffic across WAN links. All Subscribers should have a parent Subscriber, except for the first tier Subscribers that receive Distributions directly from the Distributor.

Rerouting Because of Changes to the Routing Hierarchy

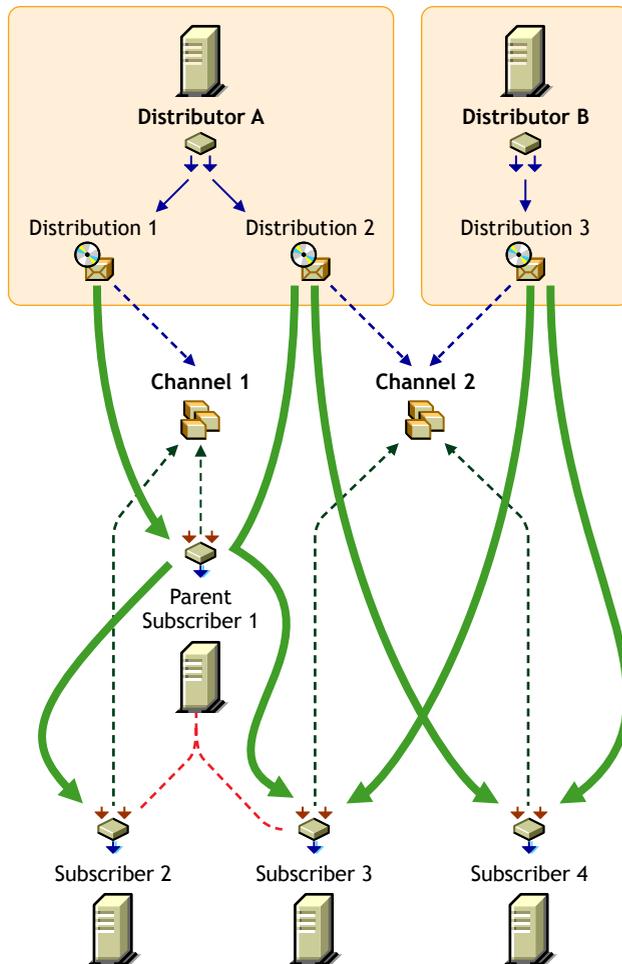
If a parent Subscriber is changed, or the routing list (on the Routing Hierarchy tab of the Distributor object's properties) is changed, the change will be reflected in the routing slip because it is calculated each time the Channel schedule starts. A refresh is required for the Distributor to re-read eDirectory and obtain the new routing hierarchy.

If a Subscriber server is removed from the network, and it was being used in a Distributor's routing hierarchy, you will need to edit the Distributor object's properties to adjust the routing hierarchy because of that Subscriber's removal. Then refresh the Distributor so it can recognize the newer routing hierarchy.

Sharing Parent Subscribers with Other Distributors

If you have multiple Distributors, they can share portions of each other's distribution routes, meaning Subscribers can be listed in the distribution routing hierarchies of more than one Distributor. This is because the route to a Subscriber is dependent on the Distributor, and can be different for any given Distributor to Subscriber path.

The following illustrates the use of multiple Distributors and parent Subscribers in sending Distributions:



The arrows and lines indicate the subscription and Distribution connections to the Channels (dotted lines) and the distribution paths from the Distributors to the Subscribers (solid lines).

Note that this illustration does not show distribution route hierarchies. For the purpose of this illustration, assume the following:

- ◆ Subscriber 1 is in Distributor A's hierarchy
- ◆ Subscriber 1 is a parent to Subscribers 2 and 3
- ◆ Subscribers 3 and 4 are in Distributor B's hierarchy
- ◆ Subscriber 4 is not in Distributor A's hierarchy

Note the following from the illustration concerning the use of multiple Distributors and parent Subscribers in sending Distributions:

- ◆ **Distribution Ownership:** Distributors have ownership of their own Distributions and will build and send each of its Distributions.
- ◆ **Multiple Distributors:** Multiple Distributors can list their Distributions in the same Channel. This means a Subscriber can receive Distributions from multiple Distributors.
- ◆ **Channel Usage by Distributors:** Distributors can list their Distributions in any Channel, and they can list one Distribution in multiple Channels.
- ◆ **Multiple Distributions per Channel:** A Channel can have multiple Distributions from one or more Distributors.

- ◆ **Channel Subscriptions:** Each Subscriber subscribes to any of the Channels that have the Distributions it needs. A Subscriber can subscribe to multiple Channels, and a Channel can have multiple Subscribers subscribed to it.
- ◆ **Parent Subscribers:** A parent Subscriber is used as a proxy for the Distributor to pass on Distributions to other Subscribers.
- ◆ **Orphaned Subscribers:** If a Subscriber is not in a Distributor's distribution route, or the child of a parent Subscriber in that hierarchy, the Distributor will send the Distribution directly to the Subscriber. This can be an issue for WAN links and other topology issues.

Distributing Across WAN Links

When you include parent Subscribers in the routing hierarchy, this can minimize network traffic by limiting the number of times a Distributor needs to pass a Distribution across a WAN link.

Because Distributors can send Distributions to parent Subscribers, which in turn can send them on to other Subscribers, a way is provided to send Distributions over a WAN link just once, instead of many times to reach every Subscriber on the other side of the WAN link.

Generally, you should have at least one parent Subscriber on every LAN to minimize the number of times a Distribution has to cross a WAN link. Even if there are only two Subscribers on a LAN, network traffic can be reduced by using one of them as the parent Subscriber.

Parent Subscribers are especially helpful with slow WAN links.

Consider the following when you determine how to distribute across your WAN links:

- ◆ **Parent Subscribers on the Distributor's LAN Segment:** You should assign at least one Subscriber to be a parent Subscriber for all of the other Subscribers on a Distributor's LAN segment. That way the Distributor can have more resources for sending Distributions across WAN links.
- ◆ **Parent Subscribers for Bridging WAN Links:** You can minimize the number of Subscribers that a Distributor must directly service across WAN links by assigning at least one parent Subscriber on all other LAN segments and including those parent Subscribers in the Distributor's routing hierarchy.

For example, your WAN has four LANs. With the Distributor in one LAN segment, it must send Distributions across three WAN links to get to Subscribers on the other three LAN segments. Let's assume each of the other LANs has 160 Subscribers who all need a Distribution from the Distributor. Without using parent Subscribers in the Distributor's routing hierarchy, the Distributor would have to send the Distribution 480 times across WAN links. In using parent Subscribers (four per LAN segment to share the Distribution workload on the LAN), the Distributor would only have to send the Distribution nine times.

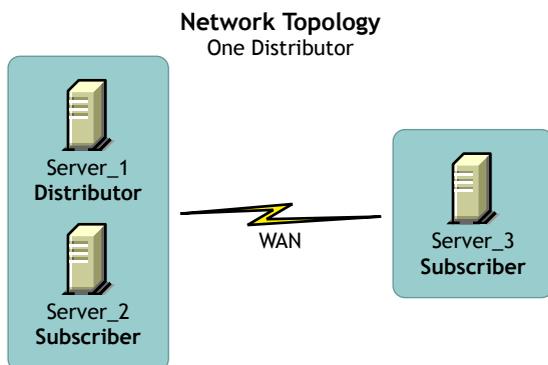
- ◆ **Primary Parent Subscribers on a LAN:** You can further minimize WAN traffic by tiering parent Subscribers on the other side of a WAN link from the Distributor. In other words, you can have just one parent Subscriber in the routing hierarchy that would also be a parent to several other parent Subscribers on its LAN segment.

Using the previous example, Subscriber 1 on each LAN segment could be the parent Subscriber for Subscribers 2, 3 and 4. In turn, parent Subscribers 1, 2, 3, and 4 would each service their own 39 or so Subscribers. That would allow the Distributor to only have to pass a Distribution across a WAN link once to Subscriber 1, which would take care of passing that Distribution on to the other three parent Subscribers, saving the Distributor three extra WAN link transmissions. Therefore, in contrast to the previous example, the 9 transmissions would be paired down to only three.

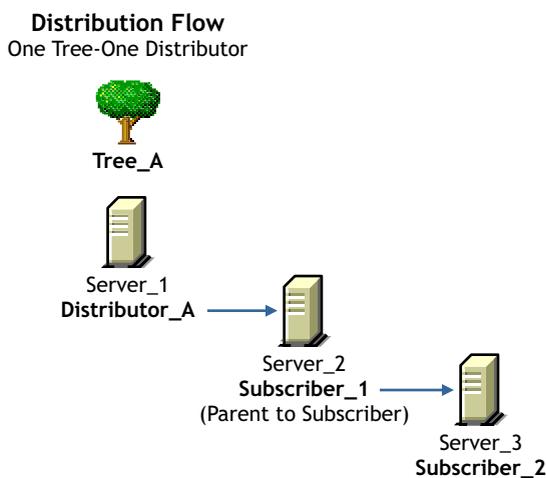
Out-of-Tree Distributions

To use Policy and Distribution Services in multiple trees, you must install the software separately in each tree. However, you only need to install the ZfS objects to one of the trees.

For example, if your network topology is:



You could have the following TED configuration for the Distributor's routing hierarchy:



In this example, the Subscriber and server objects all exist in Tree_A. This allows you to have centralized management of the TED objects, regardless of your network topology.

Although you can create the Distributor and Subscriber objects in only one tree, the Policy and Distribution Services software can be installed to any server in your network, whether the server's eDirectory object resides in the same tree where the TED objects are created, or whether the server even has an eDirectory server object in any tree (such as a Windows server in a domain). This allows you to have centralized management of TED in environments where you have multiple trees and mixed server operating systems (such as NetWare and Windows servers).

For information on how External Subscribers are used for sending Distributions between trees, see [“Sending Distributions Between Trees” on page 442](#).

Routing Hierarchy Configuration Guidelines

Parent Subscribers should be placed in the routing hierarchy using the following guidelines:

- ◆ Include at least one parent Subscriber on each LAN segment to minimize WAN traffic

- ◆ Include multiple parent Subscribers on each LAN that has 40 or more Subscribers to minimize a parent Subscriber's workload
- ◆ Make sure that every Subscriber that is not included in a Distributor's distribution route is assigned to a parent Subscriber on its LAN

Note that parent Subscribers are not always required for a WAN link. For example, if you have only two Subscribers on a LAN connected by a fast WAN link, the traffic difference between sending the Distribution once versus twice could be negligible. However, for a slow WAN link this might not be the case.

The factors in determining whether a Subscriber can receive Distributions directly from the Distributor instead of through a parent Subscriber are:

- ◆ Network connections (Within LAN? Slow or fast WAN? Across firewalls? Using NAT?)
- ◆ Frequency of distributions
- ◆ Size of the Distributions

Creating Distributors

By understanding your network's topology, your Distributions (how many, their sizes, and how often you might expect them to be rebuilt), and how many Subscribers will be receiving the various Distributions, you can determine how many Distributors you will need.

Distributors must be created by installing their software and eDirectory objects using the *ZENworks for Servers Program CD*. For more information, see [“Reinstalling ZENworks for Servers”](#) under [“Installing ZENworks for Servers”](#) in the *ZfS Installation* guide.

To determine whether you need multiple Distributors, see [“Are Additional Distributors Needed?”](#) on [page 325](#).

Configuring Distributors

Distributor objects are automatically created when the Distributor's software is installed to a server. You can edit your Distributor object's properties at any time.

Not all properties associated with the Distributor object are required. Required properties are noted in the following steps; all others are optional.

1 In ConsoleOne, right-click the Distributor object > click Properties.

2 Click General > click Settings > fill in the following fields:

Use Policy: Click to use the effective policy if you want to use the values set in the Tiered Electronic Distribution policy. This field will display if a Tiered Electronic Distribution policy has been created, distributed to the Distributor server, extracted by the Policy/Package Agent, and enforced on the server. If you enable this option, the rest of the fields are dimmed and the policy settings are used instead.

Input Rate: The rate Distributions are sent. The default is the maximum that the connection can handle. This rate is used to control a Distributor's use of narrow bandwidth links.

Output Rates Based Upon Distribution's Priority: Sets the default output rate to minimize network traffic for TED objects. This determines the send rate for Distributors. The default value is the maximum that the connection can handle. There are three output priorities where you can specify a rate:

- ◆ **High Priority:** These Distributions will be sent before any Medium or Low priority Distributions.
- ◆ **Medium Priority:** These Distributions will be sent after all High priority and before any Low priority Distributions.
- ◆ **Low Priority:** These Distributions will be sent after all High and Medium priority Distributions.

For more information, see [“Prioritizing Distributions” on page 410](#).

Maximum Concurrent Distributions: Specifies the maximum number of distribution threads that can be running concurrently. The default value is unlimited (blank field).

This number can help in load-balancing on a Distributor’s sending activity and spread network traffic over an entire scheduling window.

Connection Time-out: Specifies the allotment of time before the Distributor server times out when connecting to another node. The default value is 300 seconds (five minutes), after which it will end the connection and not retry until the send schedule starts again. The available range in seconds is 1 to 60,000.

This setting can be increased or decreased to allow messages to pass back and forth between the agents during the distribution process. If one node is expecting to receive a message from another, there should be a reasonable time to wait before assuming that the sender is no longer available.

IMPORTANT: This interval must be increased on slow or busy links where longer delays are frequent.

Working Directory: Specifies the directory to be used by the Distribution. It contains Distributions, persistent status, and temporary working files. The default path for NetWare and Windows servers is:

```
ZENWORKS\PDS\TED\DIST
```

For UNIX servers the path is:

```
usr/ZENworks/PDS/TED/Dist
```

The working directory defaults to SYS: on NetWare servers. The contents of the directory can become very large. Therefore, we recommend that you change the default from SYS: to a volume with adequate free space.

The Distributor’s working directory is also used whenever a Distribution is created. A subdirectory is created under the working directory using the DN of the Distribution object.

For more information on the working directory, see [“Working Directories” on page 447](#).

3 Click the General tab > Messaging > fill in the following fields:

Use Policy: Click to use the effective policy if you want to use the values set in the Tiered Electronic Distribution policy. This field will display if a Tiered Electronic Distribution policy has been created, distributed to the Distributor server, extracted by the Policy/Package Agent, and enforced on the server. If you enable this option, the rest of the fields are dimmed and the policy settings for messaging are used instead.

Server Console: Specifies the level of output messages to send to the Distributor console on the server console.

SNMP Trap: Specifies the level of messages to send via SNMP.

Log File: Specifies the level of messages to send to the log file.

Path and Filename: You can specify a custom log file's name and location for this Distributor object. For example:

```
ZENWORKS\PDS\TED\DIST\DISTRIBUTOR.LOG
```

The default volume is SYS: on NetWare servers. Because the log file can become quite large, we recommend that you do not use the SYS: volume.

For information on creating custom log files for all Distributor objects by using the Tiered Electronic Distribution policy, see [“Creating Custom Log Files Using Policies” on page 486](#).

Delete Log Entries Older Than __ Days: Log file entries for a Distributor will be deleted after they are older than the number of days specified. The default is six days.

E-Mail: Specifies which level of messages are sent via e-mail.

Users: Specifies e-mail users for notification.

Address Attribute: Specifies e-mail addresses for notification.

You can add users or groups stored in eDirectory or enter the e-mail addresses for users who are not contained in eDirectory. The e-mail Address Attribute associated with an eDirectory user is the default attribute.

IMPORTANT: If you select e-mail as a method for receiving notification, be aware that additional network traffic can be created.

4 Click the Schedules tab.

The schedule for a Distributor determines how often it will re-read the information contained in the TED objects in eDirectory. It reads the Channel, Distribution, and Distributor objects based on this schedule. This should be set up to reflect how often you expect information in these objects to change, or how often new objects might be created.

You can force the Distributor to re-read eDirectory by right-clicking the Distributor object and selecting the Refresh menu option.

5 Select a schedule > fill in the fields:

Use Policy: Click to use the effective policy if you want to use the values set in the Tiered Electronic Distribution policy. This field will display if a Tiered Electronic Distribution policy has been created, distributed to the Distributor server, extracted by the Policy/Package Agent, and enforced on the server. If you enable this option, the rest of the fields are dimmed and the policy settings are used instead.

Schedule Type: The Refresh schedule you selects determines when the Distributor reads eDirectory again.

IMPORTANT: We recommend the Distributor's Refresh schedule be daily, unless changes to Distributions warrant a more frequent refresh. However, do not refresh the Distributor more often than every five minutes. The following can need up to five minutes to complete their processes: Distribution building, eDirectory replication, and tree walking (when no Search policy is defined).

IMPORTANT: Changes made to TED objects (other than Distribution) are not in effect until the Distributor re-reads eDirectory.

For information on available schedules, see [Chapter 21, “Scheduling,” on page 565](#).

6 Click the Routing tab > create the Distributor's routing hierarchy.

Subscriber Routing Hierarchy: Configure the routes the Distributor will use when sending Distributions to the Subscribers. You should have planned this hierarchy in advance.

Use the following method to create the hierarchy:

6a Click the Distributor.

6b Click Add > select one or more Subscribers > click Select > click OK.

You can have multiple Subscribers directly under the Distributor.

6c Click one Subscriber.

6d Click Add > select one or more Subscribers > click Select > click OK.

You can have multiple Subscribers directly under each Subscriber.

6e Repeat **Step 6c** and **Step 6d** for each Subscriber until you have created the desired hierarchy.

7 Click the Distributions tab to view the Distributions being serviced by this Distributor.

8 To edit a Distribution, click the Distribution > click Details > edit the properties > click OK to exit the Distribution object's properties.

9 When you have finished configuring the Distributor and its Distributions, click OK to exit the Distributor object's properties.

Refreshing the Distributor

Any time you make a change in eDirectory that affects the Distributor, you must manually refresh the Distributor so that it will know of that change. The Build schedule itself only provides the Distributor with knowledge of changes to existing Distributions that it already knows about.

For example, when you create a new Distribution, the Build schedule will not make the Distributor aware of the new Distribution. You must manually refresh the Distributor so that it can detect the change in eDirectory.

To refresh the Distributor:

1 In ConsoleOne, right-click the Distributor object.

2 Click Refresh Distributor.

This causes the Distributor to re-read eDirectory and obtain all of the changes that were made in eDirectory. The Distributor Agent will then be able to act on any changes applicable to the Distributor.

To perform this task in iManager, see [“Forcing TED Agent Actions” on page 366](#).

Distribution building will begin according to the current Build schedule. The Distribution will be sent according to the Send schedule.

As soon as Subscribers receive an entire Distribution, they will extract the contents to their working directories that are specified in the Subscriber objects' properties.

Deleting a Distributor Object and How Its Distributions Are Affected

Distributor objects can be deleted from eDirectory. However, you will lose the following important information that you may want to reuse for the Distributor's replacement:

- ◆ The Distributor's distribution hierarchy that shows which Subscriber servers are used for passing on the Distributions
- ◆ The list of its Distributions (they become orphaned and unusable)

For information on how to handle orphaned Distributions, see [“Handling Orphaned Distributions” on page 411](#).

Distributions

The following sections provide concepts and instructions for the Distribution object:

- ◆ [“Understanding Distributions” on page 393](#)
- ◆ [“Distributions Issues” on page 396](#)
- ◆ [“Determining the Distributions” on page 397](#)
- ◆ [“Creating a Distribution” on page 402](#)
- ◆ [“Prioritizing Distributions” on page 410](#)
- ◆ [“Deleting a Distribution” on page 410](#)
- ◆ [“Handling Orphaned Distributions” on page 411](#)
- ◆ [“Manually Importing and Exporting Distributions” on page 412](#)
- ◆ [“Using the TED Distribution Wizard” on page 414](#)

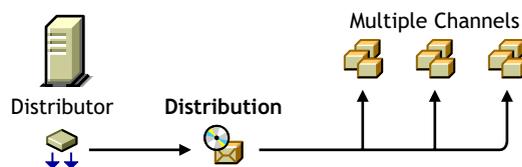
Understanding Distributions

The Distribution (TED Distribution) object contains a list of data packages or data grouping information.

- ◆ [“Functional Relationship with Other TED Objects” on page 393](#)
- ◆ [“Distribution Description” on page 394](#)
- ◆ [“Scheduling” on page 394](#)
- ◆ [“How New Versions of Existing Distributions are Created and Distributed” on page 394](#)
- ◆ [“Distribution Security” on page 395](#)
- ◆ [“Distribution Deletions” on page 395](#)

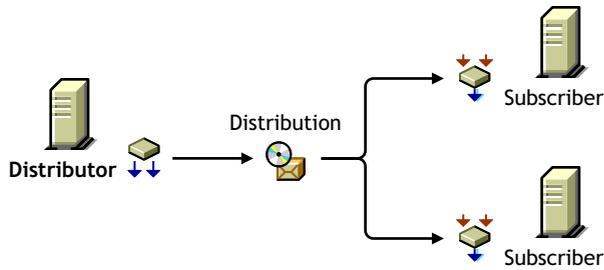
Functional Relationship with Other TED Objects

The following illustrates a Distribution’s relationship with its Distributor and the Channels:



The Distributor associates its Distributions with the Channels.

The following illustrates that a Distributor sends Distributions to Subscriber servers:



Distribution Description

A Distribution is a compilation of software and/or files, or a policy package, that the various servers in your network might need.

A Distribution is owned by only one Distributor. A Distribution keeps a list of its Channel associations, and can be placed into multiple Channels.

When a Distribution is built, it is built according to its type. There are seven types of Distributions:

- File
- FTP
- HTTP
- RPM
- Desktop Application ¹
- Policy Package
- Software Package

¹ The Desktop Application Distribution is only available when ZENworks for Desktops (ZfD) is installed.

For information on the different Distribution types, see [“The Distributions Types” on page 398](#).

Scheduling

A Distribution has a Build schedule that notifies its Distributor how often the Distribution needs to be built. If a Distribution has changed since the last time it was built, a new one will be created.

Distributions can also be made active or inactive to control whether they should be built.

For information on scheduling, see [Chapter 21, “Scheduling,” on page 565](#).

How New Versions of Existing Distributions are Created and Distributed

After you have configured a Distribution object and set the various distribution schedules, newer versions of existing Distributions are automatically created and distributed according to the following parameters:

- ◆ **Refresh Schedule:** This schedule determines when a Distributor will read eDirectory for changes to any of its Distributions. If changes are detected for a particular Distribution, it is rebuilt according to that Distribution’s Build schedule.

For more information on the Refresh schedule, see [“Distributor Object’s Refresh Schedule” on page 575](#).

- ♦ **Build Schedule:** This schedule is set independently for each Distribution. When the schedule starts for a Distribution that has been determined to have had changes to it, the Distributor proceeds to rebuild that Distribution.

For more information on the Build schedule, see [“Distribution Object’s Build Schedule” on page 575](#).

- ♦ **Maximum Revisions:** This field (in the Distribution object’s properties, on the General/Settings tab), determines how many versions of a Distribution will be kept on the Distributor and Subscriber servers’ file systems. For some Distribution types, this field determines whether a partial Distribution (delta) or complete Distribution is rebuilt. Otherwise, this field is used mainly to control disk space usage.

For more information on the Maximum Revisions field schedule, see [“Maximum Revisions” on page 396](#).

These parameters determine when a Distribution needs to be rebuilt. The other schedules (Send and Extract) determine when the rebuilt Distribution file is sent and extracted.

Distribution Security

Policy and Distribution Services provides several means for securing Distributions:

- ♦ [“Certificates” on page 395](#)
- ♦ [“Encryption” on page 395](#)
- ♦ [“Inter-Server Communications” on page 395](#)

Certificates

A certificate is a security mechanism used by Policy and Distribution Services to ensure that the Distribution received by a Subscriber was actually sent by the Distributor owning that Distribution. Without a matching certificate, a Subscriber cannot receive Distributions from the Distributor.

For more information, see [“Distribution Security Using Signed Certificates and Digests” on page 549](#).

Encryption

Distributions can be encrypted for when you send them outside your secure network.

For more information, see [“Distribution Security Using Encryption” on page 558](#).

Inter-Server Communications

Communications between TED components residing inside and outside your secure network can be secured by installing inter-server communications security where needed.

For more information, see [“Security for Inter-Server Communication Across Non-Secured Connections” on page 562](#).

Distribution Deletions

When a Distribution is built, any deletions in the Distribution object or on the Distributor server’s file system, such as deleting files or directories, will cause those files or directories to also be deleted from the Distribution when it is rebuilt. However, synchronization must be enabled in order for the files and folders to also be removed from the Subscriber server’s file system.

Distributions Issues

Consider the following in determining your Distributions:

- ◆ File sizes and their potential for compression—.JPG files won't benefit as much from compression as text files
- ◆ The bandwidth of WAN links
- ◆ The frequency of file changes
- ◆ Network resource constraints, such as low disk space or extra bandwidth availability

The better you can determine this type of information, the better you will be able to balance resource usage and minimize the use of resources.

Distributions can be configured to copy only files that are different than the target, or copy all files in their original state.

The following sections provide information about Distributions:

- ◆ [“Maximum Number of Concurrent Distributions” on page 396](#)
- ◆ [“Maximum Revisions” on page 396](#)
- ◆ [“I/O Rate \(Bytes per Second\)” on page 397](#)
- ◆ [“Updating the Distributor’s eDirectory Information” on page 397](#)
- ◆ [“Checking the Distribution Package Changes” on page 397](#)

Maximum Number of Concurrent Distributions

This is an attribute found in the Distributor and Subscriber objects. It is used to control the number of Subscribers that can be serviced concurrently when sending Distributions. This is helpful if the Distributor or parent Subscriber is servicing a large number of Subscribers. It prevents the Distributor from spreading itself very thin and sending the Distribution to all of the Subscribers at once.

For example, if a Distributor or parent Subscriber sends to 100 Subscribers and the number of concurrent Distributions is set to 10, then the sender will start with 10 connections. As one connected Subscriber finishes receiving the Distribution, another Subscriber is added in its place in the list of 10. This continues until all 100 have been serviced.

Maximum Revisions

Each Distribution allows you to determine how many versions of the Distribution will be kept by the Distributor and Subscribers in their working directories. The default is infinite for all Distribution types (except File and Desktop Application, which is 10 for both), so make sure you fill in the Maximum Revisions field attribute when creating Distributions. Consider disk space availability when calculating the maximum number of revisions.

If you enter 1, the Delete Previous Revision Before Receiving Next field becomes accessible. This allows you to control disk space by only maintaining one copy of a Distribution on the server’s file system.

The File Distribution only builds a complete Distribution the first time it creates the Distribution. All subsequent versions are just the differences (deltas) between a current version and its previous version. However, when the File Distribution reaches its maximum number of revisions, it will

delete all previous versions and build an entirely new Distribution (called a baseline), and start from 1 in counting the number of revisions.

When the maximum number of revisions is met for FTP, HTTP, and Server Software Package Distribution types, the agent will delete the oldest version of the Distribution and add the current version to the revisions. Therefore, it never exceeds the maximum number entered in the Distribution object.

I/O Rate (Bytes per Second)

This is an attribute found in the Distributor and Subscriber objects. It is used to control the amount of bandwidth used by the Distributor or parent Subscriber when sending Distributions. The default is unlimited, meaning the sender will use all the bandwidth available in sending Distributions.

Updating the Distributor's eDirectory Information

The Distributor must be updated with the configuration information contained in the TED objects in eDirectory.

Configuration changes include any changes made to the attributes of the Distributor object, Distribution objects belonging to that Distributor object, or Channel objects to which the Distributor object is associated.

The Distributor has a schedule that determines how often it reads eDirectory for configuration information. Set this schedule to coincide with the frequency at which TED objects are modified in eDirectory.

You can also force an eDirectory refresh by right-clicking a Distributor object and selecting the Refresh menu option, or by using the Zfs Management role in iManager (see [“Forcing TED Agent Actions” on page 366](#)).

Checking the Distribution Package Changes

The Distribution's schedule tells the Distributor the frequency at which the Distribution should be checked for changes.

For example, the Distribution schedule might specify a weekly build. The Distributor will rebuild that package and compare it to the previous version to see if there have been any changes.

Determining the Distributions

You can distribute whatever you can represent on the file system. This includes server applications and files. For example, the applications or files could fulfill one of the following purposes:

- ◆ Installing server software (such as virus protection software)
- ◆ Updating server software (such as a NetWare support pack)
- ◆ Updating files (such as virus patterns) on servers
- ◆ Enforcing standardization of server files or configurations (such as replacing the AUTOEXEC.NCF file on a NetWare server with an updated version)

Use a descriptive method for naming the Distributions. These names can be used to key the naming of Channels. For example:

```
VirusProtect  
VProtectPatterns
```

NW51patch4
NW6patch1
AUTOEXECNCF000326

The following sections explain the different Distribution types and issues related to determining your Distributions:

- ◆ [“The Distributions Types” on page 398](#)
- ◆ [“Determining the Sizes and Frequencies for Distribution Packages” on page 401](#)

The Distributions Types

There are several TED Distribution types. Each type has unique features that tailor it for specific needs.

- ◆ [“File” on page 398](#)
- ◆ [“FTP” on page 399](#)
- ◆ [“HTTP” on page 400](#)
- ◆ [“RPM” on page 400](#)
- ◆ [“Software Package” on page 400](#)
- ◆ [“Desktop Application” on page 401](#)
- ◆ [“Policy Package” on page 401](#)

For information on how to configure each Distribution type, see [“Creating a Distribution” on page 402](#) (specifically, [Step 6 on page 403](#)).

For the File and FTP types of Distributions, a Distribution Wizard is available for automating the process of creating them. For more information, see [“Using the Distribution Wizard”](#) under [“Installing on NetWare and Windows Servers”](#) in [“Installing Policy and Distribution Services on NetWare and Windows Servers”](#) in the *Installation* guide.

File

With this type you can select files and/or directories from the Distributor server’s file system for distribution, and select a destination location for extraction on the Subscriber.

The File type is sequential, meaning it controls the order for the building and extraction of Distributions. This prevents the building and extracting processes from being performed out of sync.

IMPORTANT: UNIX* file systems are case sensitive to allow paths and filenames that are identical except for case differences. However, if you select two such files, only the first file selected during extraction will be distributed, because the File type is not case sensitive. Therefore, do not place two files into a File Distribution where their paths and filenames are identical except for case differences.

By default, Cache and Forward is used. This process allows a parent Subscriber to begin sending a Distribution to subordinate Subscribers before it has finished receiving the Distribution. This allows entire Distributions to be sent more quickly through a chain of parent Subscribers in the Distributor’s routing hierarchy than if they each had to wait until each Subscriber had completed receiving the Distribution before it started sending.

The File Distribution is useful for distributing large Distributions that change often, thus requiring updates that need to be distributed frequently.

For the first version of a Distribution, the Distributor builds the entire Distribution (creating a baseline). A unique feature of the File type is that for all subsequent versions it calculates the differences at build time and only builds a delta of the Distribution.

The File type does this by keeping a list of the files and directories contained in a Distribution on the source machine (the Distributor or a parent Subscriber). If a source file changes, a new Distribution is built the next time its Build schedule starts. However, this new Distribution only contains the files that are different between the previous version and the current version. This is known as a delta of the original Distribution.

This delta of the Distribution file is what is distributed to the Subscribers—not the entire Distribution.

The File type is also effective when changes are frequent because it can build much smaller deltas.

There is no option to send the entire File Distribution. However, once the maximum number of revisions has been met, the Distribution will be completely rebuilt and all deltas and previous revisions will be deleted. Therefore, if you set the maximum number of revisions to 1, deltas will not be used and the entire Distribution will be built and sent every time.

For example, the first build will be the baseline Distribution (version 1), the first update (Delta 1) will be version number 2, the second update (Delta 2) will be version number 3, and so on until the number of revisions you set is reached, which triggers a new baseline rebuild. By default, this number is 10.

The maximum number of revisions can be set in the Distribution object.

If synchronization is enabled, the File type can be used for removing files and directories from the Subscriber server's file system upon extraction of the Distribution in one of two ways:

- ◆ **Edit the Distribution object:** Remove files from the list of files and directories in the Distribution object. When the Distribution is built again, those files and directories will not be included.
- ◆ **Remove files from the Distributor's file system:** Remove files from the Distributor's file system that were part of the Distribution. When the Distributor is refreshed, it will rebuild the Distribution without those files and directories.

In both cases, upon extraction of the Distribution, and with synchronization enabled, those files and directories will be removed from the Subscriber server's file system.

To manually force a Distribution to be built, you can use iManager (see [“Forcing TED Agent Actions” on page 366](#)).

FTP

With this type you can create a Distribution consisting of files from one or more FTP sources. Each source can contain one or more directories and/or files.

When an FTP site directory entry is a directory, all of its files and subdirectories are built for the Distribution.

Whenever a Distribution's Build schedule starts:

- ◆ The FTP type creates a new Distribution only if the new version would be different than the previous version.
- ◆ The Distributor builds the entire new Distribution.
- ◆ The Distributor sends each new version of the Distribution to the appropriate Subscribers.

A maximum number of revisions can be set in the Distribution object to conserve disk space. By default, the number is unlimited.

HTTP

With this type you can create a Distribution consisting of one or more HTTP sources. Each source can contain one or more target entries.

Whenever a Distribution's Build schedule starts:

- ◆ The HTTP type creates a new Distribution only if the new version would be different than the previous version.
- ◆ The Distributor builds the entire new Distribution.
- ◆ The Distributor sends each new version of the Distribution to the appropriate Subscribers.

A maximum number of revisions can be set in the Distribution object to conserve disk space. By default, the number is unlimited.

RPM

You can distribute any Red Hat Package Manager (RPM) packages that you have previously created to your Linux and Solaris servers using the RPM Distribution.

For Solaris, RPM must first be installed on the server, because it is not installed with Solaris software by default.

Whenever a Distribution's Build schedule starts:

- ◆ The Distributor builds the entire new Distribution.
- ◆ The Distributor sends each new version of the Distribution to the appropriate Subscribers.

A maximum number of revisions can be set in the Distribution object to conserve disk space. By default, the number is unlimited.

Software Package

A Server Software Package is created in ConsoleOne in the Server Software Package namespace. For more information, see [Chapter 18, "Server Software Packages," on page 493](#).

Software Package is the most robust type of Distribution. It includes installation prerequisites, pre-installation instructions, post-installation instructions, and the ability to modify text fields, SET parameters, registry settings, and the PRODUCTS.DAT file.

With the Software Package Distribution you can select .CPK files for distribution. This allows you to place a software product into a Distribution for automatic installation on the receiving server. This can include software updates to existing server software on the server.

Multiple .CPK files can be selected for one Distribution. Then, individual .CPK files will be applied on the Subscriber, depending on whether the .CPK file's prerequisites are met.

IMPORTANT: The order that the .CPK files are applied on a server is not guaranteed, and .CPK files contained in one Distribution that may start in a certain order might not all finish in that same order. Therefore, place each .CPK file in its own Distribution if you want them to be installed in a particular order and use Distribution scheduling to determine the order. For more information, see ["Forcing the Software Package Distribution Order" on page 495](#).

Desktop Application

Distributes the Application objects (that are created in ZENworks for Desktops) and the application's associated files to specified locations on the eDirectory tree and target Subscriber servers. This Distribution type allows you to solve geographic, workload, and redundancy issues for applications distributed by Novell Application Launcher that otherwise might require much of your time in manual configuration work in ZENworks for Desktops. For more information, see [Chapter 19, "Desktop Application Distribution," on page 525](#).

The Desktop Application Distribution type is not supported for Linux and Solaris servers.

This Distribution type automatically distributes a modified copy of the original Application object to a context in the eDirectory tree (a Subscriber's working context), and automatically copies the application's files to the Subscriber server that can locally service its users and workstations. It performs all of the appropriate hookups to the modified Application object to render it fully functional.

For the Desktop Application Distribution, you can set the maximum number of revisions in the Distribution object. When the version number reaches the number that you set, the Distributor rebuilds the entire Distribution. By default, this number is 10.

You can send Desktop Application Distributions to Subscriber servers on a tree that is different from the Distributor server's. However, the recipient server's Subscriber object must reside in the same tree as the modified Application objects that are created by the Distribution. The External Subscriber object is used on the Distributor's tree to send a Desktop Application Distribution to a server on another tree.

Policy Package

This type provides the mechanism for applying policies to servers. In previous versions of Policy and Distribution Services, policies were enforced through eDirectory object and container associations. With ZfS 3.0.2, policies are now distributed Subscriber servers for enforcement using the Distributed Policy Package. However, policies for Distributors continue to be enforced through context associations using the Container Package or Service Location Package.

With the Policy Package Distribution, you send policies directly to servers as Distributions, which are extracted on the receiving Subscriber server. The contained policies are then enforced on that server.

A maximum number of revisions can be set in the Distribution object to conserve disk space. By default, the number is unlimited.

For more information on each policy, see ["Server Policy Descriptions" on page 461](#).

Determining the Sizes and Frequencies for Distribution Packages

A Distribution's size and frequency of being built and sent depends on the following:

- ◆ The size and number of files being distributed. Knowing this helps in determining the amount of disk space that will be used on Distributor, Subscriber, and parent Subscribers.
- ◆ A Software Package Distribution (.CPK) always builds an entirely new version of the Distribution each time the source changes.
- ◆ HTTP and FTP Distributions always build an entirely new version of the Distribution whether the source has changed or not.
- ◆ How often the packages will change and need updating. Knowing this will help to determine how frequently new versions of the package will be created. Servers required to rebuild large

Distribution packages on a regular basis should have the processing power to perform this work. The creation of many versions of a package will also affect the amount of disk space used in the Distributor's working directory.

- ◆ The number of versions of a Distribution package that will be retained. This also affects disk space usage on the Distributor's and Subscribers' servers.
- ◆ The File Distribution creates a delta file for each new version of the Distribution until it reaches the number you have specified in the Maximum Number of Revisions field (10 is the default). Then it begins a new baseline Distribution. The delta file contains only the differences between the last and current versions of the Distribution.

Creating a Distribution

- 1** In ConsoleOne, select the container where you want the Distribution to be created > click File > click New > click Object > select the TED Distribution type > click OK.

- 2** Enter a Distribution name.

IMPORTANT: Periods (.) are not allowed in Distribution names. Instead, use dashes (-) or underscores (_) as word separators. If you use a period in the Distribution name, the Distribution will not be sent, and the Distributor will not reload after it has been exited.

- 3** To give the Distributor ownership of the Distribution, browse to select the Distributor object > click Define Additional Properties > click OK.

The Distribution object's properties are displayed.

Each Distribution belongs to a single Distributor that will build and send the Distribution.

- 4** Click the General tab > fill in the Settings tab fields:

Active: Required. In order to make a Distribution available to Subscribers, it needs to be active.

Use Digests: Digests are used by Distributors and Subscribers to verify that Distributions have not been tampered with while in transit. The digest provides a checksum for the Subscriber to compare.

Creating a digest takes more time on larger Distributions. The number of minutes per megabyte is dependent on the hardware configuration of the server where the digest is being created.

Encrypt: You can have the Distribution encrypted if you will be sending it across non-secured connections. Encryption provides security for the Distribution during transit between the Distributor and Subscriber when they are not within the same firewall. Click either Strong or Weak encryption. You also must have the same version of NCI 2.4 installed to each of these servers for encryption to work (see [“Installing NCI 2.4” on page 341](#)).

Maximum Revisions: This number helps you to control disk space usage by determining how many versions of a particular Distribution are kept in the Distributors' and Subscribers' working directories. The default is 10 for the File Distribution, and infinite for all of the other types. Increase the number if data is changing often and the changes are minimal (smaller delta files). Decrease the number if data is not changing very often, or if a significant amount of data is changing (larger delta files). If you select 1, the Delete Previous Revision field will be checked.

Delete Previous Revision Before Receiving Next: This option is available if you selected 1 as the number for the Maximum Revisions field. If the Distribution is so large that it might compromise the available disk space on the Subscriber server, you can conserve disk space by

checking this option, which will cause the previous version to be deleted before receiving the next version. If you leave the check box empty, the new version will be received in its entirety before the older version is deleted. Either way, you will have only the one version of the Distribution in the Subscriber's working directory after the Distribution has been received.

Priority: You can give the Distribution a priority that determines how it will be sent in relation to other Distributions. A High priority means it will be sent before Medium or Low priority Distributions. For information on prioritizing Distributions, see [“Prioritizing Distributions” on page 410](#).

Distributor: The DN of the Distributor object that will build and send this Distribution. This attribute cannot be modified. You selected the Distributor when you created the Distribution object.

Description: Enter useful details about the Distribution, such as the name of the desktop application, the files and directories it contains, intended user groups, and so on.

- 5 Click the General tab > click Restrictions > select a platform restriction:

Platform Restrictions: If you want to select specific operating system versions as a prerequisite to receiving this Distribution, uncheck No Restrictions and select the desired operating system version. You can select from the following:

- No Restrictions
- NetWare All
- NetWare 4.x (ZfS 2)
- NetWare 5.0 (ZfS 2)
- NetWare 5.1
- NetWare 5.x
- NetWare 6.x
- Windows Server
- Solaris
- Linux

No Restrictions means that the Distribution can be sent to any platform.

If you select NetWare All, you do not need to select any of the individual NetWare platforms.

- 6 Click the Type tab > fill in the fields:

Select Type: The type determines the type of Distribution. This field has a drop-down box where you can select the type. The options are:

- ♦ **File:** Use this option when a Distribution consists of files on the Distributor's file system that are to be copied to a Subscriber server's file system.

Use the following buttons to create the Distribution's file structure:

Button	Explanation
New Target	<p>The target file system's location for where you want the Distribution to be extracted.</p> <p>%DEST VOLUME% is the default.</p> <p>You can use any type of variable, or the actual location names. For example:</p> <p>NetWare: SYS:\FILES DATA:\FILES</p> <p>Windows: C:\FILES \\MyServer\Files (<i>shared folder</i>)</p> <p>Linux or Solaris: /usr/files</p> <p>Do not use a UNC path or all Distributions will be sent to that one location.</p>
Add Directory	<p>New Directory is the default name, which you should change to the directory name you want at the target location. Be sure to press Enter after typing the directory name, or the change will not be saved.</p> <p>Use this button to create the desired directory structure on your target Subscriber's file system for the files and directories you will be adding.</p>
Add Files	<p>Browse for directories or files on the Distributor's file system that you want copied to the target Subscriber's file system.</p> <p>Each directory or file you select will be displayed with the full path that it has on the source file system. This path identifies where to obtain the directory or file for copying to the target file system. The only path that is created on the target file system is the one you create using the New Target and Add Directory buttons, and any directories that you select with the Add Files button to add under them.</p> <p>If you select a directory, all files and subdirectories under it will also be selected for copying. Unlike the Copy File component in the Server Software Package, you cannot prune files and subdirectories from a selected directory. Any directory you browse for and add will not be expandable. You can only remove items listed in the tree structure in the Files To Be Distributed box.</p>
Delete	<p>Only deletes whatever you have selected from the tree structure in the Files To Be Distributed box.</p> <ul style="list-style-type: none"> ◆ File: Removes the file from the tree structure (not from its hard disk location). ◆ Directory: Removes the directory and any of its files and subdirectories from the tree structure. ◆ Volume: Removes all directories and files below it from the tree structure.

The File type has the following fields:

Field	Explanation
Synchronize Directories	<p>This causes the directories on the target server to be synchronized with the directories contained in the Distribution.</p> <p>WARNING: If the target server contains directories not contained in the Distribution, those directories and all files and subdirectories will be deleted from the target server's file system when the Distribution is extracted.</p> <p>This can be very destructive, especially if the target directory is a root directory. Only enable directory synchronization where you are certain you want to allow existing directories not contained in the Distribution to be deleted.</p> <p>Also, if the Distributor whose files system you are using for this Distribution is also a Subscriber that is subscribed to the Distribution, the Distributor's file system will be treated the same as the other target Subscribers.</p>
Verification Distributions	<p>Each time a Distribution changes, such as files are modified or added, a new version is built and subsequently sent to the Subscribers. However, Subscribers might need to verify that the files contained in a Distribution have been extracted and installed to all Subscribers, even when there is no new version to send.</p> <p>The verification option allows you to specify that when the Send schedule starts, if there is no new version of the Distribution to send, the Distributor should send a request for the Subscriber to re-extract the current version to ensure that the files are installed.</p>
Retry ___ Times	<p>Retries overwriting a locked file the number of times you select before failing to replace the file. Leave this check box unchecked to not replace locked files on the target file system.</p>
Kill Conn on Open Files	<p>Attempts to kill the connection of locked files so they can be overwritten. This applies only to files being extracted, not to files being accessed to build the Distribution. If a file belonging to a Distribution is locked when the Distribution is being built, the build will fail.</p> <p>Also, server and NLM™ connections cannot be killed.</p>
Maintain Trustees	<p>Maintains each file's trustee attributes for the target file system as they are on the source file system.</p> <p>This is additive, meaning it will not remove trustees on the target file system.</p>
Error Handling	<p>You have two options:</p> <ul style="list-style-type: none"> ◆ Fail on Error: The Distribution stops, allowing you to fix the error before re-sending it. This is the default option. ◆ Continue on Error: The Distribution continues with only the failed part not being finished.

- ◆ **FTP:** With this type you can create a Distribution consisting of files from one or more FTP sources. Each source can contain one or more directories and/or files.

If a target file is found to be locked during extraction, the Subscriber will throw an exception stating that the file could not be copied. The Distributor will receive this information from the Subscriber and log the failure in the reporting database.

Use the following buttons to create the Distribution's file structure:

Button	Explanation
New FTP Source	In the FTP File Group dialog box, enter the server name, a login name (the default is "anonymous"), and a password for this FTP Distribution.
New Target	Enter a volume. The variable %DEST VOLUME% is the default.
Add Directory	Browse for the directory where the file resides. If the directory has parent directories, they will all be included. You can add multiple directories. When entering information into a field, such as a directory name, be sure to press Enter or the change will not be saved.
Add Files	Browse for the files. You can add multiple files.
Delete	Deletes whatever you have selected: <ul style="list-style-type: none"> ◆ File: Removes the file from the tree (not from the FTP location). ◆ Directory: Removes the directory and any of its files and subdirectories from the tree. ◆ Volume: Removes all directories and files below it from the tree.
Properties	Displays the properties of the selected FTP source.

The FTP type has the following fields:

Field	Explanation
Files To Be Distributed	An expandable tree structure showing paths and filenames.
Binary Transfer	Enables file transfers in binary.

- ◆ **HTTP:** With this type you can create a Distribution consisting of one or more HTTP sources. Each source can contain one or more target entries.

If a target file is found to be locked during extraction, the Subscriber will throw an exception stating that the file could not be copied. The Distributor will receive this information from the Subscriber and log the failure in the reporting database.

Use the following buttons to create the Distribution's file structure:

Button	Explanation
New Target	Enter a volume. The variable %DEST VOLUME% is the default.
Add Directory	Browse for the directory where the file resides. If the directory has parent directories, they will all be included. You can add multiple directories. When entering information into a field, such as a directory name, be sure to press Enter or the change will not be saved.
Add Files	Enter the URL of the file. You can add multiple files.

Button	Explanation
Delete	Deletes whatever you have selected: <ul style="list-style-type: none"> ◆ File: Removes the file from the tree (not from the HTTP location). ◆ Directory: Removes the directory and any of its files and subdirectories from the tree. ◆ Volume: Removes all directories and files below it from the tree.

The HTTP type has the following field:

Field	Explanation
Files To Be Distributed	An expandable tree structure showing paths and filenames.

- ◆ **RPM:** Any Red Hat Package Manager (RPM) packages you have created can be distributed to your Linux or Solaris servers through TED.

Use the following buttons to add RPMs to the Distribution:

Button	Explanation
Up / Down	Arranges the order that the RPM packages will be installed.
Add From Distributor	Browse the Distributor's file system and select the RPM packages.
Add From FTP Site	Browse the FTP site and select the RPM packages.
Delete	Deletes the selected RPM package from the list.

The RPM type has the following fields:

Field	Explanation
Selected Packages	Lists the RPM packages you have added.
Installation Parameters	Lists the RPM installation parameters you have added.

- ◆ **Desktop Application:** Use this option when the Distribution consists of a ZENworks for Desktops (ZfD) application.

If a target file is found to be locked during extraction, the Subscriber will throw an exception stating that the file could not be copied. The Distributor will receive this information from the Subscriber and log the failure in the reporting database.

Use the following button to create the Desktop Application Distribution:

Button	Explanation
Setup	Starts the Desktop Application Distribution Wizard.

After exiting the wizard, the following fields and options are available:

Field	Explanation
Current Configuration	Displays the current configuration of the Desktop Application Distribution. This same information is displayed on the Summary page of the Desktop Application Distribution Wizard.

Also, the Setup button is renamed to:

Button	Explanation
Modify	Click to open the Desktop Application Distribution Wizard, where you can change the displayed configuration.

- ◆ **Software Package:** Use this option when the Distribution consists of one or more software packages created in the Server Software Package namespace in ConsoleOne.

For instructions on converting older .SPK and .CPK files to ZfS 3.0.2, see [“Converting Older Server Software Packages to ZfS 3.0.2” on page 521](#).

Use the following buttons to add software packages to the Distribution:

Button	Explanation
Up / Down	Rearranges the order that the software packages will be installed.
Add	Adds a software package to the Distribution.
Delete	Deletes the software package from those listed.

The Software Package type has the following field:

Field	Explanation
Selected Software Packages	Lists the software packages to be distributed and the order of distribution.

- ◆ **Policy Package:** Use this option when the Distribution consists of one or more policy packages containing enabled and configured policies. This is how Subscribers receive policies.

For information on creating specific policies, see [Chapter 17, “Server Policies,” on page 455](#).

Use the following buttons to add policy packages to the Distribution:

Button	Explanation
Up / Down	Rearranges the order that the policy packages will be installed.
Add	Adds a policy package to the Distribution.
Delete	Deletes the policy package from those listed.
Properties	Displays the properties of the selected policy package.

The Policy Package type has the following field:

Field	Explanation
The Following Policy Packages Will Be Distributed	Lists the policy packages to be distributed and the order of distribution.

When entering information into a field, such as a directory name, be sure to press Enter or the change will not be saved.

IMPORTANT: For the FTP, HTTP, RPM, Software Package, and Desktop Application types of Distributions, if a target file is found to be locked during extraction, the Subscriber will throw an exception stating that the file could not be copied. The Distributor will receive this information from the Subscriber and log the failure in the reporting database.

7 Click the Schedule tab > select a schedule:

The Build schedule determines how often the Distributor will build a new version of the Distribution.

Send Distribution Immediately After Building: Click this check box if you want the Distribution to be sent immediately, rather than the next time any schedules allow. However, the Subscriber's Extract schedule will determine when it is extracted for use.

Build Schedule for File Distributions: This type builds a new Distribution and compares it with the previous version for changes. If there are changes, the File type builds a file consisting of the differences between the current version and the previous version. When the maximum number of versions is reached, the type will build a complete Distribution (not just a file containing the differences) and delete all previous versions.

Build Schedule for HTTP, FTP, and Software Package Distributions: These types build new versions of the Distribution each time the Build schedule starts, regardless of whether the Distribution has changed. It will send this new version to all Subscribers.

When sending a Distribution, the sender will retry every 2 minutes for 30 minutes, then stop. It will not begin sending again until the Channel schedule starts again.

8 Click the Channels tab > fill in the field:

Channels: Each Distribution must be associated with at least one Channel if it will to be sent to a Subscriber. A Distribution will be sent to all Subscribers of the selected Channel or Channels.

9 Click OK > click Yes to resolve the certificates.

This will copy the security certificates from the Distributor to Subscriber that is subscribed to the Channel.

For information on resolving certificates, see [“Resolving Certificates” on page 552](#).

Prioritizing Distributions

Distributions can be prioritized in two ways:

- ◆ **Send Queue:** You can prioritize the order in which Distributions are sent: High, Medium, or Low. For example, in a given Channel, all High priority Distributions are sent first, then the Medium priority Distributions are sent, and then the Low priority Distributions are sent.

Because Distributions with mixed priorities cannot be sent concurrently, you can control the order in which Distributions are sent by the priorities that you assign them.

- ◆ **Output Rate:** You can configure different output rate settings for a Distribution, based on a priority: High, Medium, or Low. This allows you to control the bandwidth a Distribution will use. For example, if you want your High priority Distributions to utilize the most bandwidth, you would configure their output rates with the High priority.

The Maximum Number of Concurrent Distributions value is affected by prioritizing. This value is subordinate to the priorities set for the Distributions. For example:

- ◆ You have the concurrent Distribution number set to 10.
- ◆ There are 3 High priority Distributions.
- ◆ There are 6 Medium priority Distributions.
- ◆ There are 20 Low priority Distributions.
- ◆ Initially, only the 3 High priority Distributions will be sent concurrently.
- ◆ After all 3 of the High priority Distributions are sent, the 6 Medium priority Distributions are sent concurrently.
- ◆ After all 6 of the Medium priority Distributions are sent, 10 of the 20 Low priority Distributions are sent concurrently, and so on.

Deleting a Distribution

If you delete a Distribution object, you must immediately refresh the Distributor that owned the Distribution; otherwise, the following can happen:

- ◆ When the Build schedule fires, the Distributor will try to build a Distribution that it thinks still exists, causing an error.
- ◆ In iManager, if you click the Distribution Information option for the deleted Distribution, the Distributor will receive a 601 null-pointer error.

By immediately refreshing the Distributor, you will prevent both of these errors from occurring, because:

- ◆ The Distributor will read eDirectory when it is refreshed and no longer know of the deleted Distribution.
- ◆ The Distribution Information option for the deleted Distribution will no longer be available in iManager.

If you delete a Distribution object, you should also clean up the temporary files for the Distribution from the working directories for both the Distributor server and every Subscriber server where the Distribution was sent. You will need to do this manually on the Distributor server. You should also do this manually on each Subscriber server. You can also create a Server Software Package to automatically remove these files on the Subscriber servers.

Handling Orphaned Distributions

The following sections explain how to handle the Distributions of a deleted Distributor object:

- ◆ [“Orphaned Distributions” on page 411](#)
- ◆ [“Cleaning Up Orphaned Distributions” on page 411](#)
- ◆ [“Re-Creating Deleted Distributions” on page 411](#)

Orphaned Distributions

Because Distributions belong exclusively to their Distributors, you will no longer be able to build and send those Distributions if you delete a Distributor object from eDirectory. The Distributions associated with the deleted Distributor will become orphaned and no longer usable.

Any orphaned Distributions that have already been sent and extracted before you delete the Distributor object will be usable by the Subscriber servers where they were extracted. However, these servers will no longer receive updated versions of the orphaned Distributions.

You will still be able to see the orphaned Distribution objects in eDirectory, but no current or future Distributor object can be associated with these orphaned Distribution objects.

Cleaning Up Orphaned Distributions

For all Distribution types, you can delete the Distribution directories on the Subscriber servers’ file systems for all orphaned Distributions. We recommend that you delete the Distribution directories for any Distributions that you intend to re-create.

For most Distribution types, deleting the orphaned Distributions’ directories is all you need to do in order to clean up for management and disk space conservation purposes. These Distribution types are:

- File
- FTP
- HTTP
- RPM
- Desktop Application

However, for the Policy Package and Software Package Distribution types, you might need to undo the processes that the Distributions initiated when they were extracted and installed.

For example, a Policy Package Distribution might require that you use iManager to remove the policies that the Distribution set for the server. For more information, see [Step 5](#) under [“Managing the Policy/Package Agent from the Remote Web Console” on page 366](#).

Re-Creating Deleted Distributions

You need to re-create each orphaned Distribution that you want to continue to use. You can do this using an existing Distributor object, or after you install a new Distributor.

After you have re-created a Distribution, all Channels previously associated with the orphaned Distribution need to be associated with the newly created Distribution.

In re-creating the Distributions, you can use the configuration information from the orphaned Distribution objects. When you no longer need the orphaned Distribution objects, you can delete them and they will no longer be displayed on the Distributions tab of the Channel object.

Manually Importing and Exporting Distributions

Exporting and importing are useful for:

- ◆ Sending a large Distribution to Subscriber servers that are across a slow WAN link from the Distributor server.
- ◆ Sending a large Distribution to a parent Subscriber server that is across a slow WAN link, then having that parent pass the Distribution on to its subordinate Subscribers on its side of the WAN.
- ◆ Archiving Distributions, and later importing them when and where they are needed again.

The following sections provide information on exporting and importing Distributions:

- ◆ [“Understanding the Exporting and Importing Processes” on page 412](#)
- ◆ [“Setting Up Specialized Schedules” on page 412](#)
- ◆ [“Exporting a Distribution” on page 413](#)
- ◆ [“Importing a Distribution” on page 413](#)

Understanding the Exporting and Importing Processes

You can manually export a Distribution from a Distributor server by writing to a media source, such as a floppy disk, ZIP disk, CD, or DVD, then you can import it from that media to a Subscriber server.

The export process copies Distribution information to a UNC path or drive mapping, such as a hard drive, floppy disk, or ZIP disk. From the copy on the hard drive, you can then burn the information onto a CD or DVD.

The Distribution information includes the Channel and Distribution data from their eDirectory objects, and the content of the Distribution’s file. The Distribution information is copied to a *filename*.TED file that you name when running the Manual Distribution Wizard. You should use the .TED extension with the filename. You should also use a descriptive filename so that you can recognize the Distribution when reviewing the media content.

When the exported .TED file is imported, the eDirectory object information and the Distribution’s content are used to create the Distribution on the Subscriber server’s file system. Thereafter, deltas of the Distribution can be sent over the wire, because they are usually much smaller than the original Distribution that was exported and imported.

Distributions can only be exported and imported within the same tree where the associated Channels are known to all Distributors and Subscribers involved.

Setting Up Specialized Schedules

Depending on when you want imported Distributions to be extracted, you might want a different set of schedules set up before exporting the Distribution.

For example, if you want the exported Distribution to be extracted at different times by different Subscribers where it is imported, then:

- 1** Set the build schedule for the Distribution to be exported to Immediate.
- 2** Add the Distribution to a Channel with a Send schedule set to Never.
This prevents Subscribers that have not yet had the Distribution manually imported to them from receiving a Channel's notice to trigger extraction of the yet-to-be-received Distribution.
- 3** Add all of the Subscribers where the Distribution is to be imported to the Channel you used in [Step 2](#).
- 4** Refresh the Distributor that owns the Distribution to be exported.
- 5** After the Distribution has been built, continue with [“Exporting a Distribution” on page 413](#).

If you do not need a specialized schedule, you can just follow the instructions in the next two sections, which assume that existing schedules are acceptable.

Exporting a Distribution

- 1** In ConsoleOne, click Tools, then click Manual Distribution to start the Manual Distribution Wizard.
- 2** Click Export, then click Next.
- 3** Select a Channel, select a Distribution from that Channel, then click Next.
This Channel's ID is retained in the .TED file for use when importing the Distribution.
- 4** For the Distribution, provide a path (UNC or drive mapping) and filename (descriptive for identifying which Distribution is on the media), then click Next.
The filename should have .TED as its extension.
- 5** If you are satisfied with the summary, click Finish.
The .TED file is saved to the path that you specified.
- 6** If your path was to a hard drive, you can now burn the .TED file to a CD or DVD.

Importing a Distribution

- 1** In ConsoleOne, click Tools, then click Manual Distribution to start the Manual Distribution Wizard.
- 2** Click Import, then click Next.
- 3** Provide the path and filename to the .TED file, then click Next.
This is the .TED file that you exported to a media source.
- 4** Select parent Subscribers in the top box and individual Subscribers in the bottom box, then click Next.

If you select a parent Subscriber that is in the routing hierarchy, all of the Subscribers below it in the hierarchy have the imported Distribution passed on to them, but only if they are already subscribed to the Distribution's Channel.

The Subscribers displayed in the bottom box are those who are currently subscribed to the Distribution's Channel. The heading displays the Channel that is associated with the Distribution being imported. This information is contained in the .TED file being imported.

External Subscribers are not listed in the bottom box because they cannot receive manual Distributions.

- 5** If you are satisfied with the summary, click Finish.

The Distribution is copied from the media source you specified and placed in the working directories of the selected Subscribers. The Channel and Distribution objects' information is written to eDirectory.

At this point, imported Distributions are not viewable in iManager.

- 6** If you set up specialized schedules for the imported Distribution (see [“Setting Up Specialized Schedules” on page 412](#)), restart the ZENworks Server Management process on each Subscriber server where it was imported; otherwise, skip to [Step 7](#).

The Distribution is extracted on the Subscriber servers according to their individual Extract schedules. After extraction, you can view the Distribution's information in iManager.

- 7** To make the Subscribers recognize that they have received a new Distribution:

- 7a** Under the ZfS Management role in iManager, click Remote Web Console.

- 7b** Identify the Distributor owning the imported Distribution in either of the following fields:

IP Address or DNS Name

Distributor, Subscriber, or Server Object Name

- 7c** Click OK.

- 7d** In the Display field, select Tiered Electronic Distribution.

- 7e** Click the Channels tab, then select Distribute Channel.

- 7f** Click the Channel associated with the imported Distribution, then click OK.

The Subscribers can now act as though they had received the Distribution across the wire. The Distributor begins to send the Distribution listed in the Channel to the Subscribers, but the Subscribers reply that they already have the Distribution, then begin to extract it.

If a Subscriber is a parent Subscriber that needs to pass the imported Distribution on to subordinate Subscribers, it does so when the Distribution's Channel starts.

Using the TED Distribution Wizard

ZfS provides the TED Distribution Wizard to help you learn the process involved in creating and sending a Distribution. This wizard can be used to create and send either a File or FTP Distribution.

To use the TED Distribution Wizard:

- 1** In ConsoleOne, click the container where you want the Distribution object created > click Tools > TED Distribution Wizard.
- 2** Review the information on the Introduction page > click Next.
- 3** On the Distributor Selection page, browse for and select the Distributor that will own this File or FTP Distribution > click Next.
- 4** On the Subscriber Selection page, click Add > browse for the Subscribers that will receive this Distribution > click Select > click OK > click Next.
- 5** On the File Source page, select the file source (the Distributor's file system, or a remote FTP site) > click Next.
- 6** On the Destination Volume or Drive page, select an option and fill in its field > click Next.

Use the Same Volume or Drive for All Subscribers: If each target Subscriber will have the exact same volume or drive available, select this option and enter the volume label or drive letter.

Use a Variable for the Volume or Drive: If your target Subscribers will be using different paths (for example you have NetWare, Windows, and UNIX Subscriber servers), you can enter a variable value. This value must be defined on each Subscriber in order to receive the Distribution.

- 7** On the Additional Destination Directories page, enter any additional path information for the target Subscriber servers > click Next.

Note that your path information is displayed under the "Data Will Be Placed In Path" heading as you type it. Use this information to verify that the path is valid before continuing.

- 8** On the File Selection From Distributor Server page, click Add > browse for the files or directories to be included > click Select > click OK > click Next.

You are browsing the Distributor's file system, not the local machine's.

Repeat clicking Add until you have all of the files and directories you want in this Distribution.

- 9** On the Distribution Name and Context page, fill in the fields > click Next.

Distribution Name: Enter a unique name for the Distribution.

Context: Browse for and select the container where you want the Distribution object to be created.

- 10** On the Additional Options page, check or uncheck the options as applicable > click Next.

The following options are all enabled by default:

Copy the Distributor's Security Certificate To All Subscribers: This is necessary for the Subscriber to be able to receive and extract this Distribution. This might not be necessary if you run the wizard again with the same Distributor and Subscribers.

Verify That All Subscribers Are Up and Running: If you want to make sure your target Subscribers will be able to receive this Distribution, check this option.

Notify the Distributor To Read eDirectory For New Information: This will cause the Distribution to be built immediately.

- 11** On the Summary page, review the steps that will be take by the TED Distribution Wizard > click Finish to create the Distribution.

Information will be displayed as the Distribution is created and sent.

- 12** To review the log file, click Yes when prompted.

If you click Yes, you can review the log file. Click Close to exit the log window and the TED Distribution Wizard.

If you click No, the TED Distribution Wizard is exited.

Channels

The following sections provide concepts and instructions for the Channel object:

- ◆ ["Understanding Channels" on page 416](#)
- ◆ ["Creating and Configuring Channels" on page 417](#)

- ◆ [“Forcing a Channel To Be Sent” on page 418](#)

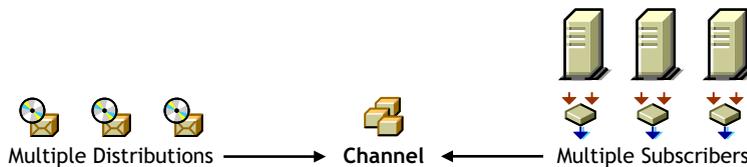
Understanding Channels

The Channel object (TED Channel) contains a list of Distributions associated with it and Subscribers subscribed to it.

- ◆ [“Functional Relationship with Other TED Objects” on page 416](#)
- ◆ [“Channel Description” on page 416](#)
- ◆ [“Scheduling” on page 416](#)
- ◆ [“Subscriptions to Channels” on page 416](#)

Functional Relationship with Other TED Objects

The following illustrates a Channel’s relationship with Distributions and Subscribers:



The Distributions are listed in the Channel, and the Subscribers subscribe to the Channel.

Channel Description

Distributors can list one Distribution in multiple Channels, and multiple Distributors can list their Distributions in the same Channel.

You can have as many Channels as you want. Channels do not hold the actual Distributions—only a reference to them. There is no limit to the number of Distribution references a Channel can send. The practical limit is how many Distributions you want to track per Channel.

Scheduling

A Channel’s Send schedule determines when a Distribution can be sent from the Distributor to its Subscribers.

A Channel can be active or inactive to control when its Distributions can be sent.

For information on how time zones can affect scheduling between a Channel and its associated Distributors and Subscribers, see [“TED Object Scheduling Issues” on page 568](#).

Subscriptions to Channels

Channels can be subscribed to by multiple Subscribers.

To receive a Distribution, a Subscriber must subscribe to the Channel where that Distribution is listed. However, a Subscriber will receive all of the Distributions listed in that Channel, which means they will all be applied to the Subscriber server when they are extracted.

Creating and Configuring Channels

The following sections provide you with the steps to create and configure the TED objects with ConsoleOne.

Do the following in order for each Distributor:

- ◆ “Determining the Channel Names” on page 417
- ◆ “Creating the Channel Objects” on page 417
- ◆ “Configuring the Channels” on page 417

Determining the Channel Names

In naming Channels, use a descriptive method. For example:

```
VirusProtect  
VProtectPatterns  
VirusProtection  
NW51patch4  
NW6patch1  
AUTOEXECNCNF000326
```

You will be able to manage your Channels more easily by:

- ◆ Using names that are purpose oriented
- ◆ Using a similar name for the Channel and its Distributions

Creating the Channel Objects

Channels are used to group Distributions and establish a schedule for passing a Distributor’s Distributions to Subscribers that are subscribed to the Channel. A Channel can have Distributions from many Distributors. A Channel can be subscribed to by many Subscribers.

To create a Channel object:

- 1** In ConsoleOne, select a container object to hold the Channel object > click File > New > Object > Channel.
- 2** Provide a name for the Channel object > click OK.
- 3** Create as many Channel objects as needed to group Distributions by type and/or send schedule.

Configuring the Channels

You need to configure a Channel object before you can begin using it.

Not all properties associated with the Channel object are required. Required objects are noted; all others are optional.

To configure the Channel object:

- 1** In ConsoleOne, right-click the Channel object > click Properties.
- 2** Click the General tab > fill in the fields:
Active: Click the check box to enable the Channel to pass on its Distributions.

Description: Provide a useful description, such as what Distributions the Channel is associated with.

3 Click the Distributions tab > click Add to add Distributions.

Distributions: List of Distributions that are associated with this Channel. For information on creating Distribution packages, see [“Distributions” on page 393](#).

4 Click the Subscribers tab > click Add to add Subscribers to the Channel.

Subscribers Subscribed to This Channel: List of Subscribers and External Subscribers that are subscribed to this Channel.

5 Click the Schedule tab > select a schedule for when to distribute the Channel’s Distributions.

For information on available schedules, see [Chapter 21, “Scheduling,” on page 565](#).

Forcing a Channel To Be Sent

If you want to send all of the Distributions in a Channel outside of Channel’s the normal Send schedule, you can manually force the distribution process.

Assuming a new Distribution has been built and the Channel’s Send schedule is not ready to fire, do one of the following to force a Channel to be sent:

- ◆ Using the ZfS Management role in iManager, click Channel > Distribute Channel.
- ◆ In ConsoleOne, you have a two-step process:
 1. Click the Channel object > click Properties > click the Schedule tab > select Run Immediately > click OK > right-click the Distributor object > click Refresh Distributor.
 2. After the Distribution has been sent, click the Channel object > click Properties > click the Schedule tab > select the schedule that the Channel previous had > click OK.

As soon as a Subscriber receives an entire Distribution, it will extract it according to the Subscriber’s Extract schedule.

Subscribers

The following sections provide concepts and instructions for the Subscriber object:

- ◆ [“Understanding Subscribers” on page 418](#)
- ◆ [“Creating Subscribers” on page 420](#)
- ◆ [“Configuring Subscribers” on page 420](#)
- ◆ [“Updating Subscriber Configurations” on page 423](#)
- ◆ [“Associating Subscribers with Channels” on page 424](#)
- ◆ [“Deleting Subscriber Objects That Are Part of a Distributor’s Routing Hierarchy” on page 424](#)

Understanding Subscribers

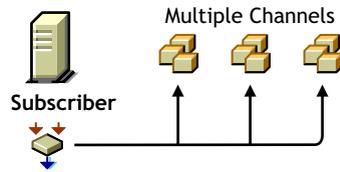
The Subscriber object (TED Subscriber) is an eDirectory object that defines the properties for the Subscriber.

- ◆ [“Functional Relationship with Other TED Objects” on page 419](#)
- ◆ [“Subscriber Description” on page 419](#)
- ◆ [“Scheduling” on page 419](#)

- ◆ “Subscribing to Channels” on page 419
- ◆ “Parent Subscribers” on page 420

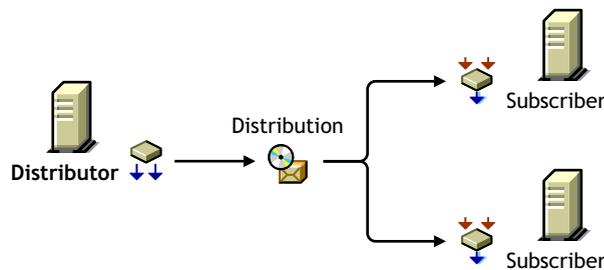
Functional Relationship with Other TED Objects

The following illustrates a Subscriber’s relationship with the Channels:



The Subscriber subscribes to the Channels.

The following illustrates the Subscriber’s relationship with Distributors and Distributions:



Subscriber Description

The Subscriber is a service that receives and extracts Distributions to obtain the software, files, or policies it needs.

Any server where you want to distribute applications, files, or policy packages must have the Subscriber software installed and a Subscriber object in the eDirectory tree. The Subscriber object can be in a different tree than the server’s NCP server object, because IP addresses or DNS names are used for moving Distribution files to the Subscriber servers.

Distributions are copied to the Subscriber server’s hard drive. The Subscriber Agent receives the Distributions and extracts them to install the software, files, or policies.

Scheduling

A Subscriber’s Extract schedule determines when it can extract its Distributions.

For information on scheduling, see [Chapter 21, “Scheduling,” on page 565](#).

Subscribing to Channels

Subscribers can subscribe to a Channel to receive all of the Distributions listed in that Channel. A Subscriber object’s properties lists the Channels it is subscribed to.

Subscribers can receive Distributions from multiple Distributors because:

- ◆ Multiple Distributors can list their Distributions in the same Channel
- ◆ Subscribers can subscribe to multiple Channels

Parent Subscribers

Subscribers can be parent Subscribers, which are proxies for the Distributor to pass Distributions on to other Subscribers. This helps the Distributor by providing load-balancing for sending Distributions to many Subscribers.

The Subscriber object's properties lists the parent Subscriber through which it receives all of its Distributions. A Subscriber can receive its Distributions directly from the Distributor if it does not have a parent Subscriber and is not listed in the Distributor's routing hierarchy.

Parent Subscribers can also be used to bridge WAN links to ensure that Distribution packages are sent across WAN links a minimum number of times.

Creating Subscribers

Subscribers must be created by installing their software and eDirectory objects using the *ZENworks for Servers Program CD*. For more information, see “[Reinstalling ZENworks for Servers](#)” under “[Installing ZENworks for Servers](#)” in the *ZfS Installation* guide.

If a Subscriber object is inadvertently deleted, you can re-create it in ConsoleOne. However, the revision number of the new Subscriber object will be less than its revision number in the ted.cfg file. Therefore, the Subscriber cannot accept any updates to its configuration, because the lower revision number causes it to assume that the configuration data is older than what it has. To resolve this problem, delete the ted.cfg file on the Subscriber server, and the next time a Distribution is sent to the Subscriber, a new configuration is accepted, and a new ted.cfg file created.

Configuring Subscribers

Subscriber objects are automatically created when you install the Subscriber software to a server.

Not all properties associated with the Subscriber object are required. Required objects are noted; all others are optional.

To configure the Subscriber object's properties:

- 1 In ConsoleOne, right-click the Subscriber object > click Properties.
- 2 Click the General tab > click Settings > fill in the following fields:

Use Policy: Click to use the effective policy if you want to use the values set in the Tiered Electronic Distribution policy. This field will display if a Tiered Electronic Distribution policy has been created, distributed to the Subscriber server, extracted by the Policy/Package Agent, and enforced on the server.

If you enable this option, the rest of the fields are dimmed and the policy settings are used instead. The current policy is displayed in parentheses.

Input Rate: The rate Distributions are received or sent. The default is the maximum that the connection can handle. This rate is used to control a Subscriber's use of narrow bandwidth links. This also defines the rate between a parent Subscriber and its subordinate Subscribers.

Output Rates Based Upon Distribution's Priority: Sets the default output rate to minimize network traffic for TED objects. This determines the send rate for Subscribers. The default value is the maximum that the connection can handle. There are three output priorities where you can specify a rate:

- ◆ **High Priority:** These Distributions will be sent before any Medium or Low priority Distributions.

- ♦ **Medium Priority:** These Distributions will be sent after all High priority and before any Low priority Distributions.
- ♦ **Low Priority:** These Distributions will be sent after all High and Medium priority Distributions.

For more information, see [“Prioritizing Distributions” on page 410](#).

Maximum Concurrent Distributions: Specifies the maximum number of distribution threads that can be running concurrently. The default value is unlimited (blank field). This applies to parent Subscribers that will pass on Distributions to subordinate Subscribers.

Connection Time-out: Specifies the number of seconds a Subscriber will wait for a response from a Distributor (receiving) or a Subscriber (sending) before ending the connection. If a connection is ended during sending or receiving, the send will not start again until the next time the Channel schedule starts. It will then pick up where it left off. The default value is 300 seconds (five minutes). The available range in seconds is 1 to 60,000. This setting should be a reasonable time to wait for a response from one node to another.

IMPORTANT: This interval must be increased on slow or busy links where longer delays are frequent.

Working Directory: Specifies the directory to be used by the Distribution. It contains Distributions, persistent status, and temporary working files. The default path for NetWare and Windows servers is:

ZENWORKS\PDS\TED\SUB

For UNIX servers the path is:

usr/ZENworks/PDS/TED/Sub

The working directory defaults to SYS: on NetWare servers. The contents of the directory can become very large. Therefore, we recommend that you change the default from SYS: to a volume with adequate free space.

For more information on the working directory, see [“Working Directories” on page 447](#).

Parent Subscriber (optional): Specifies a parent Subscriber from which Distributions can be received.

This field is where you can specify a parent Subscriber in the routing hierarchy for passing on the Distribution if you do not want the Distributor to send Distributions directly to the External Subscriber's server in the other tree.

Disk Space Desired To Be Left Free: Use this value to ensure there will be enough free disk space for receiving Distributions. A Subscriber will not attempt to receive a Distribution if the disk space value set here is insufficient.

- 3 Click the General tab > click Messaging > fill in the following fields:

Use Policy: Click to use the effective policy if you want to use the values set in the Tiered Electronic Distribution policy. This field will display if a Tiered Electronic Distribution policy has been created, distributed to the Subscriber server, extracted by the Policy/Package Agent, and enforced on the server.

If you enable this option, the rest of the fields are dimmed and the policy settings for messaging are used instead. The current policy is displayed in parentheses.

Server Console: Specifies the level of output messages to send to the Subscriber console on the server console.

SNMP Trap: Specifies the level of messages to send via SNMP.

Log File: Specifies the level of messages to send to the log file.

Path and Filename: You can specify a custom log file's name and location for this Subscriber object. For example:

```
ZENWORKS\PDS\TED\SUB\SUBSCRIBER.LOG
```

The default volume is SYS: on NetWare servers. Because the log file can become quite large, we recommend that you do not use the SYS: volume.

For information on creating a custom log file for all Subscriber objects, see [“Creating Custom Log Files Using Policies” on page 486](#).

Delete Log Entries Older Than __ Days: Log file entries for a Subscriber will be deleted after they are older than the number of days specified. The default is six days.

E-Mail: Specifies which level of messages to send via e-mail.

Users: Specifies e-mail users for notification.

Address Attribute: Specifies e-mail addresses for notification.

You can add users or groups stored in eDirectory or enter the e-mail addresses for users who are not contained in eDirectory. The e-mail Address Attribute associated with an eDirectory user is the default attribute.

IMPORTANT: If you select e-mail as a method for receiving notification, be aware that additional network traffic can be created.

- 4 Click the General tab > click Working Context > browse for a working context.

This is the eDirectory context where the Subscriber will create the objects related to the Desktop Application Distributions it receives.

- 5 Click the Schedules tab > select a schedule > fill in the fields:

Use Policy: Click to use the effective policy if you want to use the values set in the Tiered Electronic Distribution policy. This field will display if a Tiered Electronic Distribution policy has been created, distributed to the Subscriber server, extracted by the Policy/Package Agent, and enforced on the server. If you enable this option, the rest of the fields are dimmed and the policy settings for scheduling are used instead.

Schedule Type: This schedule determines when the Subscriber will extract the Distributions.

For information on available schedules, see [Chapter 21, “Scheduling,” on page 565](#).

- 6 Click the Channels tab > fill in the fields:

Active: To activate a Channel for this Subscriber server so it can receive the Channel's Distributions, click a Channel > check the box to enable it. To deactivate a Channel so that the Subscriber will not receive the Channel's Distributions, uncheck the box to disable it.

Channel: Click Add to create a Channel. Click Details to edit a Channel.

- 7 Click the Variables tab > fill in the fields:

Include Policy: Click to use the effective policy if you want to use the values set in the Tiered Electronic Distribution policy. This field will display if a Tiered Electronic Distribution policy has been created, distributed to the Subscriber server, extracted by the Policy/Package Agent, and enforced on the server.

If you click this option, the variables specified in the Tiered Electronic Distribution policy will be added to the list of variables. However, if there are duplicate variables, the variables in the Subscriber will prevail.

Variable: Name of the variable. It should indicate how the variable will be used. For example, WORKINGVOL.

Value: The value that the Subscriber will use when this variable is specified. For example, DATA:

To ensure that extraction will take place, provide an absolute path to the Subscriber. For example, if the path is only the DATA volume, make sure the colon (:) is included, because it is a necessary part of the full path.

Description: Describes how the variable will be used. For example:

`Volume for the working directory.`

For information on variables, see [“Using Variables to Control File Extraction” on page 584](#).

- 8** To include this Subscriber in a group, click Group Membership > click Add > browse for a Subscriber Group object > click Select > click OK.
- 9** When you are finished configuring the Subscriber object, click OK to exit the Subscriber object’s properties.

Updating Subscriber Configurations

The Subscriber software cannot run on a server if the Subscriber does not know its TED configuration, such as where its working directory is. Therefore, during the installation process, you determine a basic TED configuration for each of the Subscribers that you are installing.

Using this input, the installation program creates a TEDNODE.PROPERTIES file on each Subscriber server that contains the Subscriber's initial TED configuration. Until a server receives its first Distribution, this TEDNODE.PROPERTIES file provides the server with its TED configuration information, so that it can function as a Subscriber.

A Subscriber server can only receive configuration information from a Distributor server whose Distributor object is in the same tree as the server's Subscriber object. This is known as the trusted tree, which is established during the installation process. For information on when knowing the trusted tree is necessary, see [“Subscriber Software Configuration and Trusted Trees” on page 428](#).

When a Distributor server sends a Distribution to a Subscriber server, the Distributor first checks to see if that Subscriber server has a current TED configuration in the form of a TED.CONFIG file. If this is the first time the Subscriber has received a Distribution, it will not have that file. The Distributor then sends the TED.CONFIG file to the Subscriber, and the TEDNODE.PROPERTIES file is no longer used by the Subscriber. Then the Distributor checks again to see if the Subscriber server has a current TED.CONFIG file. Upon confirmation from the Subscriber, the Distribution is sent. In other words, the Distributor will never send a Distribution to a Subscriber server whose configuration information is not current.

The TED.CONFIG file can be updated any time you make configuration changes to the Subscriber object's properties. However, Subscribers do not read eDirectory, so when a change is made to the Subscriber, it must rely on the Distributor server to discover those changes and send the new configuration information to the Subscriber server, updating its TED.CONFIG file.

If you should install the Subscriber software to a server that will not have a Subscriber object in any eDirectory tree, such as a Microsoft domain server, the TEDNODE.PROPERTIES file will be used by such servers, in lieu of having its TED configuration updated by a Distributor server. In this case, for configuration changes, you would need to edit the server's TEDNODE.PROPERTIES file. For more information, see [“The TEDNODE.PROPERTIES File Requirement” on page 430](#) and [“Editing the TEDNODE.PROPERTIES File” on page 451](#).

Associating Subscribers with Channels

Before a Subscriber can receive a Distribution, you need to associate the Subscriber to a Channel. This can be done either from the Subscriber or Channel object's properties.

To associate a Channel with a Subscriber:

- 1 In ConsoleOne, right-click the Subscriber object > click Properties.
- 2 Click the Channel tab > click Add > add the needed Channels.
- 3 Click OK to save the changes.
- 4 Click the Schedule tab > select a schedule.

The schedule determines when Distributions that have been received are extracted or installed.

For information on the available schedules, see [Chapter 21, "Scheduling," on page 565](#).

- 5 Click the Variables tab > fill in the following fields > click OK:

Variable Name: Can be used to determine the location of the destination directory where files will be extracted. Enter the name of the variable exactly as you will be using it within the %...% symbols.

Value: This is the value of the variable, which can be another variable's name.

Description: Text field to enter details about the variable.

For information on variables, see ["Using Variables to Control File Extraction" on page 584](#).

Deleting Subscriber Objects That Are Part of a Distributor's Routing Hierarchy

If a Subscriber object is removed from eDirectory, or a Subscriber server is removed from the network (whether its Subscriber object is also removed or left in eDirectory), and that Subscriber was part of a Distributor's routing hierarchy, you will need to edit the Distributor object's properties to adjust the routing hierarchy accordingly. Otherwise, Distributions that were being sent through that parent Subscriber would not reach the designated Subscriber servers.

Subscriber Groups

The following sections provide concepts and instructions for the Subscriber Group object:

- ♦ ["Understanding Subscriber Groups" on page 424](#)
- ♦ ["Creating and Configuring Subscriber Groups" on page 425](#)

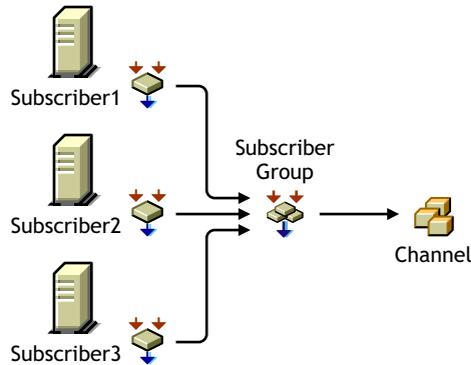
Understanding Subscriber Groups

A Subscriber Group is an eDirectory object (TED Subscriber Group) used for grouping Subscribers objects.

- ♦ ["Functional Relationship with Other TED Objects" on page 425](#)
- ♦ ["Subscriber Group Description" on page 425](#)
- ♦ ["Scheduling" on page 425](#)

Functional Relationship with Other TED Objects

The following illustrates a Subscriber Group's relationship with Subscribers and Channels.



Subscriber Group Description

A Subscriber Group is used for grouping Subscribers that have the same Distribution needs.

Subscriber Groups are useful when you will be sending several different Distributions to the same set of Subscribers. There is no need to create a Subscriber Group if it will only be associated with one Channel.

For example, Distribution A will be in Channel A, Distribution B will be in Channel B, and so on. Then, without using a Subscriber Group, you would need to subscribe each of your Subscribers to Channel A, then each to Channel B, and so on, which could be a very long process. However, by using a Subscriber Group, you will only need to create the group, add the Subscribers to it, then subscribe that one group to each Channel.

Another use of a Subscriber Group is that when the group is associated with two or more Channels, you can edit the group's membership more easily than making the same changes in multiple Channels. For example, to remove a Subscriber from one Subscriber Group, you just edit that one group's properties. To remove that same Subscriber from several Channels, you would need to edit each Channel's properties.

Scheduling

Subscriber Groups are not scheduled.

Creating and Configuring Subscriber Groups

- 1** In ConsoleOne, select the container to hold the Subscriber Group object > click File > New > Object > TED Subscriber Group.
- 2** In the New TED Subscriber Group dialog box, enter a name for the Subscriber Group (worksheet [item 17](#)) > click Define Additional Properties > click OK.
- 3** In the General Settings tab, enter a description.
- 4** To populate the group with Subscribers, click the Members tab > do the following:
 - 4a** Click Add > browse for and select the Subscriber objects (worksheet [item 18](#)) > click OK.
 - 4b** To remove any Subscribers from the list, select the Subscribers > click Delete.
 - 4c** To view the properties of any Subscriber, select the Subscriber > click Details.

5 Click OK when you have finished configuring the Subscriber Group object.

External Subscribers

The following sections provide concepts and instructions for the External Subscriber object:

- ◆ [“Understanding External Subscribers” on page 426](#)
- ◆ [“Using External Subscribers for Out-of-Tree Distributions” on page 430](#)
- ◆ [“Creating and Configuring External Subscribers” on page 432](#)

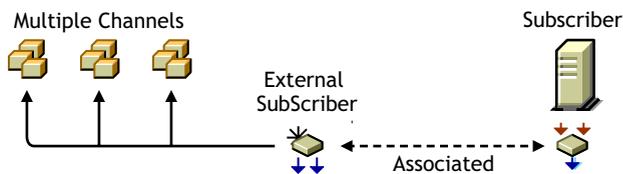
Understanding External Subscribers

An External Subscriber is an eDirectory object (TED External Subscriber) that represents a Subscriber object in another tree.

- ◆ [“Functional Relationship with Other TED Objects” on page 426](#)
- ◆ [“External Subscriber Description” on page 426](#)
- ◆ [“Subscriber Software Configuration and Trusted Trees” on page 428](#)
- ◆ [“Scheduling” on page 430](#)

Functional Relationship with Other TED Objects

The following illustrates an External Subscriber’s relationship with the Channel:



The External Subscriber subscribes to the Channels.

External Subscriber Description

A Distributor cannot send its Distributions to a Subscriber server whose Subscriber object is in a different tree than the Distributor’s object, or to a server that does not have a Subscriber object. An External Subscriber object is needed for out-of-tree distributions.

For information on the External Subscriber object, see the following:

- ◆ [“The External Subscriber’s Purpose” on page 427](#)
- ◆ [“Duplicate Distribution Management” on page 427](#)
- ◆ [“External Subscriber Characteristics” on page 427](#)
- ◆ [“External Subscriber Requirements” on page 427](#)
- ◆ [“The External Subscriber Object’s Properties” on page 428](#)

The External Subscriber's Purpose

If you installed all of your TED objects in one tree, an External Subscriber object is not necessary, because you can send your Distributions using the Distributor and Subscriber objects that are in the same tree.

However, the External Subscriber object is useful for sending out-of-tree Distributions when one of the following conditions exists:

- ♦ **The Target Server Has No Subscriber Object in Any Tree:** The target server, such as a Windows server in a Microsoft domain, has only the Subscriber software installed on it.
- ♦ **The Target Server Has a Subscriber Object in a Different Tree:** The target server has the Subscriber software installed on it, but its Subscriber object is in a different tree than the Distributor object that is sending the Distribution.

Because the External Subscriber is only an object in an eDirectory tree, it does not actually handle the Distribution files; it simply identifies which server is to receive them.

Duplicate Distribution Management

An External Subscriber object can be used to circumvent the need to duplicate Distribution work in another tree.

For example, a few Subscribers on a tree at a remote site could receive all of their Distributions via the External Subscriber in the Distributor's tree. That would prevent the need to have a Distributor server at the remote site, including duplicating the Distribution configuration and management effort there.

External Subscriber Characteristics

An External Subscriber is associated with a server running the Subscriber software that has no Subscriber object in any tree, or no Subscriber object in the same eDirectory tree as the Distributor from which it will receive the Distribution.

External Subscriber objects are associated with a Subscriber server through an IP address or DNS name of that server.

You can send Distributions outside of eDirectory, such as to a Windows server in a Microsoft domain. For more information on this type of distribution, see [“Subscriber Software Configuration and Trusted Trees” on page 428](#) and [“The TEDNODE.PROPERTIES File Requirement” on page 430](#).

External Subscriber objects cannot be parent Subscribers. If an External Subscriber has a parent Subscriber, both the External Subscriber's and parent Subscriber's objects must reside in the same tree.

External Subscriber Requirements

If a target server's Subscriber object is in a different tree from the Distributor object of the server that will send it a Distribution, that target server must be represented by an External Subscriber object in the Distributor's tree.

Because TED uses IP addresses or DNS names to locate servers, Subscriber objects can be in a different tree than those servers' NCP objects.

An External Subscriber must be subscribed to the Channel that lists the Distributions needed by its associated Subscriber.

The server receiving a Distribution via an External Subscriber must have the Subscriber software installed on it so that it can receive and extract the Distribution. It is not required to have a Subscriber object in any tree, such as if it is a Windows server in a domain (see [“Subscriber Software Configuration and Trusted Trees” on page 428](#) and [“The TEDNODE.PROPERTIES File Requirement” on page 430](#)).

The External Subscriber Object’s Properties

The External Subscriber object properties contain only the following:

- ◆ IP address or DNS name of the Subscriber server that’s in a different tree or a domain (required)
This is the ID of the Subscriber server in one tree that is to receive a Distribution from a Distributor in another tree (the tree where the External Subscriber object resides).
- ◆ The Channels it is subscribed to (required)
This is for identifying which Distributions need to be sent to the Subscriber server in the other tree.
- ◆ Membership in a Subscriber Group (optional)
This can be used for subscribing to the Channels subscribed to by the group.
- ◆ Context of a parent Subscriber in the External Subscriber’s own tree (optional)
A parent Subscriber is usually in the Distributor's distribution hierarchy.
If used, the parent Subscriber will do the physical work in sending the Distribution file to the server in the other tree. Otherwise, the Distributor server will send the Distribution directly to the Subscriber server in the other tree.

Subscriber Software Configuration and Trusted Trees

Subscribers can be configured by a Distributor, but External Subscribers cannot. External Subscribers are just objects identifying a server. However, a Subscriber server identified by an External Subscriber object must have a TED configuration in order to receive the Distributions via the External Subscriber object.

Using the External Subscriber object brings up the need to understand trusted trees:

- ◆ [“The Reason for Trusted Trees” on page 428](#)
- ◆ [“Determining the Trusted Tree” on page 429](#)
- ◆ [“The TEDNODE.PROPERTIES File Requirement” on page 430](#)

The Reason for Trusted Trees

The following applies to any NetWare or Windows server, whether it has an NCP server object in an eDirectory tree or a server object in a Microsoft domain:

- ◆ During installation, the server can have both a Subscriber object created for it and the Subscriber software installed to it
- ◆ During installation, the server can have only the Subscriber software installed to it (no Subscriber object is created)
- ◆ During installation, you should identify the trusted tree of any server that will not have a Subscriber object created for it

Identifying a trusted tree has two purposes:

- ◆ To locate a Distributor that can update the Subscriber's TED configuration information
- ◆ To indicate which tree to accept policies from

A Subscriber server's TED configuration information is stored in eDirectory in its Subscriber object (which the Distributor reads), and in a TED.CONFIG file in the Subscriber server's file system (which the Subscriber reads). A Distributor server sending the configuration information must have its Distributor object in the same tree as the Subscriber object that it is configuring.

A Subscriber server can receive its Subscriber software configuration only from a Distributor in its trusted tree. The trusted tree is where the server's Subscriber object and that Distributor object both reside. This is not the tree where an associated External Subscriber object resides, and it doesn't matter whether it's the same tree where the server's NCP object resides.

A Subscriber server that does not have a Subscriber object in any tree (such as a Windows server in a Microsoft domain), must use its TEDNODE.PROPERTIES file for its TED configuration information. This file is created on the server when you installed the Subscriber software. Then it can receive and extract Distributions from a Distributor in another tree (via an External Subscriber object). The extraction process is the time when the trusted tree requirement must be met. For more information, see [“The TEDNODE.PROPERTIES File Requirement” on page 430](#).

Determining the Trusted Tree

There are two situations that deal with whether to install Subscriber objects for Subscriber servers:

- ◆ **eDirectory Server:** When you install the Subscriber software to a server whose NCP object is in another tree, you have one of the following options:
 - ◆ You can create the Subscriber object in the Distributor's tree, which may not be the tree where the Subscriber server's NCP object resides (the server's Subscriber and NCP objects do not need to be in the same tree). In this case, you will not need an External Subscriber object for sending Distributions to that Subscriber, because its object will not be out-of-tree.

The Subscriber server's trusted tree will be the same tree where the Distributor object resides. Therefore, it will receive its TED configuration updates from the Distributor in its trusted tree.

- ◆ You can elect to not create a Subscriber object for the server. In this case, you will need to use the TEDNODE.PROPERTIES file to configure that Subscriber server. You will also need to use an External Subscriber object to send Distributions to that server.

In order for this Subscriber to have policies enforced on it, you would need to identify its trusted tree, which would be the tree it receives Policy Package Distributions from.

- ◆ **Non-eDirectory Server:** When you install the Subscriber software to a server that is in a Microsoft domain, and therefore will not have an NCP object in any eDirectory tree, you might not create a Subscriber object for this server (however, you can). Therefore, you will need to use the TEDNODE.PROPERTIES file to configure that Subscriber server. You will also need to use an External Subscriber object to send Distributions to this server.

In order for this Subscriber to have policies enforced on it, you would need to identify its trusted tree, which would be the tree it receives Policy Package Distributions from.

The File Installation Paths and Options page in the installation program contains the Trusted Tree field. However, this field will only be displayed if you uncheck the Create eDirectory Objects check box on the Installation Options page. This causes the installation program to install only software for the selected servers.

You must select a trusted tree for each server where you have selected to install the Subscriber software, or your Policy Package Distributions may not extract on that Subscriber server, because policies point to objects in a tree.

For installation instructions concerning the Trusted Tree field, see the steps in the applicable sections under [“Reinstalling ZENworks for Servers”](#) under [“Installing ZENworks for Servers”](#) in the *ZfS Installation* guide.

The TEDNODE.PROPERTIES File Requirement

A TEDNODE.PROPERTIES file must be used to provide configuration information for the following Subscriber servers:

- ◆ A Subscriber server that has a Subscriber object and has not yet received its first Distribution. After it does, it will then use the TED.CONFIG file given to it by the Distributor in its trusted tree that is sending that first Distribution, and it will no longer use the TEDNODE.PROPERTIES file.

A Subscriber can only be configured by a Distributor server whose object is in the same tree as the Subscriber’s object.

- ◆ A Subscriber server that does not have a Subscriber object in any tree.

This could be a Windows server in a Microsoft domain where you only installed the Subscriber software without creating the object.

If you installed the Subscriber software (using the ZfS installation program) without creating the Subscriber object, the TEDNODE.PROPERTIES file was automatically created and configured.

For more information, see [“Editing the TEDNODE.PROPERTIES File” on page 451](#).

Scheduling

The External Subscriber object is not scheduled.

Using External Subscribers for Out-of-Tree Distributions

Review the following sections to understand how to use External Subscribers for out-of-tree distributions:

- ◆ [“External Subscriber, One Distributor, and One Tree” on page 430](#)
- ◆ [“External Subscriber, Multiple Distributors, and Multiple Trees” on page 431](#)

External Subscriber, One Distributor, and One Tree

After you install Policy and Distribution Services software to your servers, you can send Distributions to a server that does not have a Subscriber object in any tree using the External Subscriber object.

The following TED configuration might exist for the Distributor’s routing of its Distributions through External Subscribers:

Distribution Flow One Tree-One Distributor



In this example, the Server_2 does not have a Subscriber object in any tree. It only has the Subscriber software installed on it so that it can receive and extract Distributions. It can be a NetWare server with an NCP server object in any tree, or a Windows server in a Microsoft domain.

To send a Distribution from Distributor_A to Server_2, you would create an External Subscriber object in Tree_A and list Server_2's IP address or DNS name in the External Subscriber object's properties.

The eDirectory Distribution View

From an eDirectory perspective, the Distribution is sent from the Distributor object to the External Subscriber object, which in turn sends it to Server_2. You can use a parent Subscriber in Tree_A (not shown) where you do not want the Distributor to be directly sending its Distributions to Server_2.

The Actual Distribution Process

From a topology perspective, the Distribution file is sent from Server_1 to Server_2, using the IP address or DNS name of Server_2 that is located in the External Subscriber object's properties.

Configuring the Subscriber Server

Server_2 receives its TED configuration information from the TEDNODE.PROPERTIES file installed on its server when the Subscriber software was installed there. Because there is no Subscriber object to configure, you would need to edit Server_2's TEDNODE.PROPERTIES file in order to make configuration changes. For information on editing the TEDNODE.PROPERTIES file, see [“Editing the TEDNODE.PROPERTIES File” on page 451](#).

The Subscriber Server's Trusted Tree

In order for Server_2 to have policies enforced on it, Tree_A would need to be established as its trusted tree during installation of the Subscriber software to the server. For the installation steps, see [“Reinstalling ZENworks for Servers”](#) under [“Installing ZENworks for Servers”](#) in the *Installation* guide.

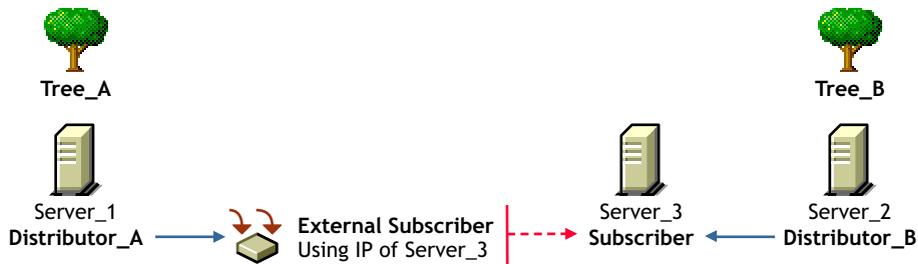
External Subscriber, Multiple Distributors, and Multiple Trees

After you install Policy and Distribution Services software to your servers in multiple trees, you can send Distributions between trees using the External Subscriber object.

The following TED configuration might exist for the Distributor's routing of its Distributions through External Subscribers:

Distribution Flow

Two Trees-Two Distributors



In this example, Server_3 has a Subscriber object in Tree_B.

To send a Distribution from Distributor_A to Server_3, you would create an External Subscriber object in Tree_A and list Server_3's IP address or DNS name in the External Subscriber object's properties.

The eDirectory Distribution View

From an eDirectory perspective, the Distribution is sent from Distributor_A to the External Subscriber object, which in turn sends it to Server_3. You can use a parent Subscriber in Tree_A (not shown) where you do not want Distributor_A to be directly sending its Distributions to Server_3.

The Actual Distribution Process

From a topology perspective, the Distribution file is sent from Server_1 to Server_3, using the IP address or DNS name of Server_3 that is located in the External Subscriber object's properties.

Subscriber Server_3's Trusted Tree and Its TED Configuration

Each tree has a Distributor that provides configuration information for the Subscriber servers in its own tree.

Server_3 will receive its TED configuration information from Distributor_B, because Tree_B was set as Server_3's trusted tree when it was made a Subscriber using the installation program. However, Server_3 cannot extract a Distribution from Distributor_A until it has been configured by Distributor_B, which is done the first time the Subscriber receives a Distribution from Distributor_B.

Creating and Configuring External Subscribers

You can create External Subscriber objects for sending Distributions to Subscriber servers with Subscriber objects residing on other trees or to Subscriber servers that do not have a Subscriber object in any tree.

The following sections provide steps to create and configure an External Subscriber:

- ◆ [“Creating an External Subscriber Object” on page 432](#)
- ◆ [“Configuring the External Subscriber Object” on page 433](#)

Creating an External Subscriber Object

To create an External Subscriber object:

- 1** In ConsoleOne, select the container to hold the External Subscriber object > click File > New > Object > TED External Subscriber.
- 2** Enter a name for the External Subscriber object.
Make the name unique to help identify the server from the other tree.
- 3** Enter the server's TCP/IP address or DNS name > click OK.
This must be a valid TCP/IP address or fully distinguished DNS name.

Configuring the External Subscriber Object

To configure an External Subscriber object:

- 1** In ConsoleOne, right-click an External Subscriber object > click Properties.
- 2** Click the General Settings tab > fill in the Setting fields:
 - Use Policy:** Click this check box if you want to use the values set in the Tiered Electronic Distribution policy that is being enforced on the External Subscriber's server.
If you enable this option, the Parent Subscriber field is dimmed and the policy settings are used instead.
 - Parent Subscriber:** Specifies a parent Subscriber from which all Distributions will be received.
Because the routing hierarchy in a Distributor object's properties only accounts for parent Subscribers, this field is where you can connect an end-node Subscriber to the routing hierarchy. These end-node Subscribers (which in this case are External Subscribers) cannot be used to pass Distributions to other Subscribers.
- 3** Click the Network Address tab > verify the IP address of the External Subscriber's server.
 - IP Address:** You entered this IP address when you created the object. Verify that it is correct.
- 4** Click the Channels tab > fill in the fields > click OK:
 - Active:** To activate a Channel for this External Subscriber server so it can receive the Channel's Distributions, click a Channel > check the box to enable it. To deactivate a Channel so that the External Subscriber will not receive the Channel's Distributions, uncheck this box to disable it.
 - Channel:** Click Add to create a Channel. Click Details to edit a Channel.
- 5** To include this External Subscriber in a group, click Group Membership > click Add > browse for a Subscriber Group object > click Select > click OK.
- 6** When you are finished configuring the External Subscriber object, click OK to exit the object's properties.

Configuring Multiple TED Objects

When you have the same configuration change to make to several TED objects, you can save time by modifying the properties of multiple objects.

You can perform multiple object modifications for the following TED objects:

Distributor
Distribution
Channel

Subscriber
External Subscriber
Subscriber Group
Policy Package

For more information, see:

- ◆ [“Issues with Modifying Multiple TED Object Properties” on page 434](#)
- ◆ [“Modifying Multiple TED Object Properties” on page 434](#)
- ◆ [“Property Tabs Available for Multiple-Object Modifications” on page 435](#)

Issues with Modifying Multiple TED Object Properties

- ◆ **Available Properties:** Although the purpose is to provide a means to make the same changes to multiple objects, not all properties for the TED objects can be modified using this method.

The Schedule and Other property tabs are not available for editing the properties of multiply-selected TED objects. For the Distribution object, the Type tab is also not available. For changes to these property tabs, you must edit each TED object individually.

- ◆ **Modified Fields:** The fields where you make changes in the Properties of Multiple Objects dialog box are the only modifications that will be made for the selected objects. In other words, if you leave a field blank (you do not modify it), no change will be made in that field for all of the selected objects. Each object will retain its original field entry.

Where objects have different information in a given field, that field is blank in the Properties of Multiple Objects dialog box.

- ◆ **Removing Information:** In some fields, a space is a valid entry. This can be used as a method for removing varied existing entries for each of the selected TED objects when you want the field to be blank for all of the selected objects.
- ◆ **Policy Defaults:** If you have a Tiered Electronic Distribution policy in force, the Use Policy check box will be displayed in each TED object’s properties, but only checked for the individual TED objects where the policy applies (because their properties have never been edited, or you enabled that check box).

For multiple object properties, if the Use Policy check box is displayed and checked, the policy’s contents will be displayed in dimmed text in the applicable fields. These attributes are only applicable to those TED objects whose individual properties contain a checked Use Policy check box.

You can uncheck the Use Policy check box when editing multiple properties to disable the Tiered Electronic Distribution policy for the selected TED objects that were previously using the policy. Any changes you make will be replicated to all selected TED objects and the Tiered Electronic Distribution policy will no longer be in force for any of those objects.

IMPORTANT: If the Working Directory field for an object received its location from the Tiered Electronic Distribution policy, and you disable the Use Policy check box when editing multiple properties, the Working Directory field will then be left blank for that object. Therefore, the next time you access the properties for that object, you will be required to enter a working directory location.

Modifying Multiple TED Object Properties

To modify the properties of multiple TED objects:

- 1 In ConsoleOne, select a number of TED objects.

They must be of the same type, such as all Distributor objects. The Properties of Multiple Objects menu option will not display if you select multiple objects of different types.

You can select multiple objects using the Shift and Ctrl keys.

2 Right-click the selected objects > click Properties of Multiple Objects.

Each of the selected objects will be listed in the Objects to Modify tab on the Properties of Multiple Objects dialog box. These are the objects that will have their properties modified when you make changes.

3 To change the objects displayed in the list, click Add or Remove.

The Add button allows you to browse for other TED objects. Only objects of the type you have previously selected will be displayed for adding to the list.

Before clicking the Remove button, you must first select one or more objects in the list. This only removes the objects from the list, not from eDirectory.

4 Click a tab containing the property that you want to modify.

For descriptions of the property tabs available for the various TED objects, see [“Property Tabs Available for Multiple-Object Modifications” on page 435](#).

5 Edit the property.

The changes will be made to all of the objects listed in the Objects to Modify tab.

For more information on individual property fields, see the descriptions within the steps in the following sections:

- ◆ [“Configuring Distributors” on page 389](#)
- ◆ [“Creating a Distribution” on page 402](#)
- ◆ [“Creating and Configuring Channels” on page 417](#)
- ◆ [“Configuring Subscribers” on page 420](#)
- ◆ [“Creating and Configuring Subscriber Groups” on page 425](#)
- ◆ [“Creating and Configuring External Subscribers” on page 432](#)

6 Repeat [Step 4](#) and [Step 5](#) until you have finished modifying the various properties for the selected objects.

7 When finished modifying properties, click OK to close the Properties of Multiple Objects dialog box.

All changes that you have made will be updated for all of the selected objects.

Property Tabs Available for Multiple-Object Modifications

The following tables list the property tabs that are available in the multiple object editing mode for each TED object.

IMPORTANT: Generally, if you change information, it will be changed for all of the selected objects. Exceptions are noted in the explanations.

- ◆ [“Distributor Object” on page 436](#)
- ◆ [“Distribution Object” on page 436](#)
- ◆ [“Channel Object” on page 437](#)
- ◆ [“Subscriber Object” on page 437](#)

- ◆ “External Subscriber Object” on page 438
- ◆ “Subscriber Group Object” on page 439
- ◆ “Policy Package Object” on page 439

Distributor Object

Property Tabs Available	Explanation
Objects to Modify	You can add or remove Distributor objects from the list of objects to be modified. This does not add or remove the objects from eDirectory.
General	<p>This includes the Settings and Messaging subtabs.</p> <p>For the Settings subtab, none of the fields will display information, even if it is identical between the selected Subscriber objects. However, dimmed text will be displayed in fields where the Tiered Electronic Distribution policy is in effect for one or more of the selected TED objects.</p> <p>In the Settings subtab, you can only add new information that will be applied to all of the selected Subscriber objects. In the Messaging subtab, you can edit existing entries.</p>
Routing	If there are any differences in routing hierarchies between the selected Distributor objects, nothing will be displayed for this tab. You can only edit routing hierarchies for multiple Distributor objects when they are identical.
NDS Rights	This tab includes the New Trustees and the Inherited Filter Rights subtabs.

Distribution Object

Property Tabs Available	Explanation
Objects to Modify	You can add or remove Distribution objects from the list of objects to be modified. This does not add or remove the objects from eDirectory.
General	<p>This includes the Settings and Restrictions subtabs.</p> <p>For the Settings subtab, none of the fields will display information, even if it is identical between the selected Subscriber objects. However, dimmed text will be displayed in fields where the Tiered Electronic Distribution policy is in effect for one or more of the selected TED objects.</p> <p>In the Settings subtab, you can only add new information that will be applied to all of the selected Subscriber objects. In the Restrictions subtab, you can edit existing entries.</p>
Channels	<p>Channels do not automatically display on this tab. You can only browse for Channels to add to each of the selected Distribution objects, or browse for a Channel to be removed from each of the selected Distribution objects that are associated with that Channel.</p> <p>Adding or removing a Channel in the list on this tab does not add or remove the Channel object from eDirectory.</p>
NDS Rights	This tab includes the New Trustees and the Inherited Filter Rights subtabs.

Channel Object

Property Tabs Available	Explanation
Objects to Modify	You can add or remove Channel objects from the list of objects to be modified. This does not add or remove the objects from eDirectory.
General	<p>This includes the Settings subtab (with the Active check box and the Description field).</p> <p>For the Settings subtab, none of the fields will display information, even if it is identical between the selected Subscriber objects. However, dimmed text will be displayed in fields where the Tiered Electronic Distribution policy is in effect for one or more of the selected TED objects.</p> <p>In the Settings subtab, you can only add new information that will be applied to all of the selected Subscriber objects.</p>
Distributions	<p>Distributions do not automatically display on this tab. You can only browse for Distributions to add to each of the selected Channel objects, or browse for a Distribution to be removed from each of the selected Channel objects that are associated with that Distribution.</p> <p>Adding or removing a Distribution in the list on this tab does not add or remove the Distribution object from eDirectory.</p>
Subscribers	<p>Subscribers do not automatically display on this tab. You can only browse for Subscribers to add to each of the selected Channel objects, or browse for a Subscriber to be removed from each of the selected Channel objects that are associated with that Subscriber.</p> <p>Adding or removing a Subscriber in the list on this tab does not add or remove the Subscriber object from eDirectory.</p>
NDS Rights	This tab includes the New Trustees and the Inherited Filter Rights subtabs.

Subscriber Object

Property Tabs Available	Explanation
Objects to Modify	You can add or remove Subscriber objects from the list of objects to be modified. This does not add or remove the objects from eDirectory.
General	<p>This includes the Settings, Messaging, and Working Context subtabs.</p> <p>For the Settings subtab, none of the fields will display information, even if it is identical between the selected Subscriber objects. However, dimmed text will be displayed in fields where the Tiered Electronic Distribution policy is in effect for one or more of the selected TED objects.</p> <p>In the Settings subtab, you can only add new information that will be applied to all of the selected Subscriber objects. In the Messaging subtab, you can edit existing entries.</p>

Property Tabs Available	Explanation
Channels	<p>Channels do not automatically display on this tab. You can only browse for Channels to add to each of the selected Subscriber objects, or browse for a Channel to be removed from each of the selected Subscriber objects that are associated with that Channel.</p> <p>Adding or removing a Channel in the list on this tab does not add or remove the Channel object from eDirectory.</p>
Variables	<p>You can only add a new variable for all of the selected objects. Variables that are common among all of the selected objects are not displayed for editing. You must visit each Subscriber object individually to modify existing variables.</p>
Group Membership	<p>Group Memberships do not automatically display on this tab. You can only browse for Group Memberships to add to each of the selected Subscriber objects, or browse for a Group Membership to be removed from each of the selected Subscriber objects that are associated with that Group Membership.</p> <p>Adding or removing a Group Membership in the list on this tab does not add or remove the Group Membership object from eDirectory.</p>
NDS Rights	<p>This tab includes the New Trustees and the Inherited Filter Rights subtabs.</p>

External Subscriber Object

Property Tabs Available	Explanation
Objects to Modify	<p>You can add or remove External Subscriber objects from the list of objects to be modified. This does not add or remove the objects from eDirectory.</p>
General	<p>This includes the Settings subtab.</p> <p>For the Settings subtab, only the Parent Subscriber field exists. If you make an entry here, all selected External Subscribers will have the same parent Subscriber.</p>
Channels	<p>Channels do not automatically display on this tab. You can only browse for Channels to add to each of the selected External Subscriber objects, or browse for a Channel to be removed from each of the selected External Subscriber objects that are associated with that Channel.</p> <p>Adding or removing a Channel in the list on this tab does not add or remove the Channel object from eDirectory.</p>
Group Membership	<p>Group Memberships do not automatically display on this tab. You can only browse for Group Memberships to add to each of the selected Subscriber objects, or browse for a Group Membership to be removed from each of the selected Subscriber objects that are associated with that Group Membership.</p> <p>Adding or removing a Group Membership in the list on this tab does not add or remove the Group Membership object from eDirectory.</p>
NDS Rights	<p>This tab includes the New Trustees and the Inherited Filter Rights subtabs.</p>

Subscriber Group Object

Property Tabs Available	Explanation
Objects to Modify	You can add or remove Subscriber Group objects from the list of objects to be modified. This does not add or remove the objects from eDirectory.
General	This includes the Settings and Messaging subtabs.
Channels	<p>Channels do not automatically display on this tab. You can only browse for Channels to add to each of the selected Subscriber Group objects, or browse for a Channel to be removed from each of the selected Subscriber Group objects that are associated with that Channel.</p> <p>Adding or removing a Channel in the list on this tab does not add or remove the Channel object from eDirectory.</p>
Group Members	<p>Group Members do not automatically display on this tab. You can only browse for Group Members to add to each of the selected Subscriber objects, or browse for Group Members to be removed from each of the selected Subscriber objects that are associated with that Group Membership.</p> <p>Adding or removing a Group Membership in the list on this tab does not add or remove the Group Membership object from eDirectory.</p>
NDS Rights	This tab includes the New Trustees and the Inherited Filter Rights subtabs.

Policy Package Object

Property Tabs Available	Explanation
Objects to Modify	You can add or remove Policy Package objects from the list of objects to be modified. This does not add or remove the objects from eDirectory.
Policies	This includes the various supported platform subtabs. For more information on the policies available on these platforms, see “Server Policy Descriptions” on page 461 .
Distributions	<p>Distributions do not automatically display on this tab. You can only browse for Distributions to add to each of the selected Policy Package objects, or browse for a Distribution to be removed from each of the selected Policy Package objects that are associated with that Distribution.</p> <p>Adding or removing a Distribution in the list on this tab does not add or remove the Distribution object from eDirectory.</p>
NDS Rights	This tab includes the New Trustees and the Inherited Filter Rights subtabs.

Sending Distributions

For information on sending Distributions, see the following:

- ◆ [“Understanding the Distribution Processes” on page 440](#)
- ◆ [“Forcing a Single Distribution To Be Sent” on page 441](#)

- ◆ [“Sending Distributions Through Parent Subscribers” on page 441](#)
- ◆ [“Sending Distributions Between Trees” on page 442](#)

Understanding the Distribution Processes

Following are the processes for creating and sending a Distribution, generally done in this order:

1. **Configure and schedule the Distributors.** You must use the installation program on the *ZENworks for Servers Program* CD to create a Distributor.
For information on Distributors, see [“Distributors” on page 379](#) and [“Distributor Object’s Refresh Schedule” on page 575](#).
2. **Configure and schedule the Subscribers.** You must use the installation program on the *ZENworks for Servers Program* CD to create a Subscriber.
One of the primary configurations that you must do for Subscribers is to associate them with the Channels that hold the Distributions they need. For more information, see [“Associating Subscribers with Channels” on page 424](#).
For information on Subscribers, see [“Subscribers” on page 418](#) and [“Subscriber Object’s Extract Schedule” on page 577](#).
3. **Configure the necessary policies.** Policy Packages that contain the desired policies must be created in ConsoleOne or iManager before they can be distributed.
For information on policies, see [“Configuring Server Policies” on page 467](#).
4. **Create, configure, and schedule the Distributions.** You can use either ConsoleOne or iManager to create Distribution objects.
This could be the most time-consuming portion of the whole process, depending on the complexity of the Distribution to be configured. After you have set up your Distributors and Subscribers and create the Distribution objects, you only need to utilize the Distributors’ routing hierarchies for distributing the files and policies to your Subscriber servers.
The Distribution object’s schedule is the best place to prevent an individual Distribution from being sent.
For information on Distributions, see [“Distributions” on page 393](#) and [“Distribution Object’s Build Schedule” on page 575](#).
5. **Create, configure, and schedule the Channels.** You can use either ConsoleOne or iManager to create Channel objects.
Usually, you will create a new Channel for each Distribution. It is generally easier to manage your distribution system by matching Channels with what they distribute. However, you can include multiple Distributions in a Channel, such as when they are related and all Subscribers subscribing to the Channel will need all of those Distributions. For example, a Channel could hold several Distributions that each contain a different virus pattern update.
The Channel object is normally the best object to use for controlling whether Distributions should be sent. Setting its schedule to Never effectively stops the distribution process for all of the Distributions listed in it.
For information on Channels, see [“Channels” on page 415](#) and [“Channel Object’s Send Schedule” on page 576](#).

The Distributions are built, sent, and extracted according to the schedules that you set for each of the TED objects involved.

For information on the distribution processes, see [“The Basic Distribution Process” on page 372](#).

You may have accomplished some of the above processes during installation of ZfS and during your initial system configuration (see [Chapter 14, “Configuring Policy and Distribution Services,” on page 317](#)).

Forcing a Single Distribution To Be Sent

If you want to send a single Distribution outside of the normal Refresh, Build, and Send schedules, and the Channel’s Send schedule is not ready to fire, you can manually force this distribution process using only the ZfS Management role in iManager.

To force a single Distribution to be sent, do one of the following:

- ◆ If the Send Distribution Immediately After Building option is checked in the Distribution’s properties, in iManager click Distribution > click Build Distribution.

Even if there are other Distributions in the Channel where this Distribution is listed, only this Distribution will be sent.

- ◆ If the Send Distribution Immediately After Building option is not checked in the Distribution’s properties, in iManager click Distribution > click Build Distribution > click Channel > Distribute Channel.

All other Distributions in the Channel will be also be sent if needed by the Subscribers.

As soon as a Subscriber receives an entire Distribution, it will extract it according to the Subscriber’s Extract schedule.

Sending Distributions Through Parent Subscribers

Subscribers can not only receive and extract Distributions, they can also pass on Distributions to other Subscribers. Subscribers that pass on Distributions are known as parent Subscribers.

Parent Subscribers do not need to be subscribed to the Distributions they are passing on. They simply receive a Distribution for passing it on to a subordinate Subscriber that has done two things:

- ◆ Subscribed to the Channel listing the Distribution
- ◆ Identified the parent Subscriber in the subordinate Subscriber’s object properties

To set up parent Subscribers for passing on Distributions:

1 Determine a Subscriber object (hereafter referred to as "child Subscriber") that cannot receive a certain Distribution because this child Subscriber is not contained in the Distributor’s routing hierarchy (the Distributor owning this Distribution).

2 In that Subscriber object’s properties, click the General tab > click Settings > in the Parent Subscriber field browse for a Subscriber object that is contained in the Distributor’s routing hierarchy > click OK.

This establishes the Subscriber selected as a parent Subscriber. This distinction is not kept in the parent Subscriber’s object properties, but only in the child Subscriber’s.

3 Create a Channel object where only the child Subscriber is associated.

4 Create a Distribution > associate it with the child Subscriber’s Channel.

5 Send this Distribution.

Because this Distribution is associated only with the Channel where the child Subscriber is subscribed, the parent Subscriber will not extract it, but only pass it on to the child Subscriber.

Because the parent Subscriber is in the routing hierarchy of the Distributor, it will have access to the Distribution for passing it on. Because the child Subscriber does not have any access to the Distributor, it needed the parent Subscriber to provide access to the Distribution.

Although you can establish a parent Subscriber for a child Subscriber, the child Subscriber can still be subscribed to a Channel where the parent Subscriber is subscribed. Both Subscribers can receive and extract that Channel's Distributions without the parent Subscriber passing it on to the child Subscriber, because the child can have access to that particular Distributor's routing hierarchy. The key is whether the Distributor owning the desired Distribution can send it to the child Subscriber without using a parent Subscriber.

Sending Distributions Between Trees

Using External Subscribers, you can send Distributions from one tree to another. To accomplish this, do the following:

- 1** Make sure TED is installed to both trees.

In the remaining steps, TREE1 represents the tree where the Distribution is created and TREE2 represents the other tree where you want the Distribution sent.

The server in TREE2 that is to receive the Distribution from TREE1 must have the Subscriber software installed on it (meaning it is a Subscriber in TREE2).

For information on installing TED, see [“Getting Started”](#) in the *Installation* guide.

- 2** In TREE1, create an External Subscriber object.

Make sure that the IP address or DNS name you enter for this object matches the Subscriber server in TREE2 where you want the Distribution to be sent.

For steps in creating External Subscribers, see [“Creating and Configuring External Subscribers”](#) on page 432.

- 3** In TREE1, create the Channel for the Distribution.

For steps in creating Channels, see [“Creating and Configuring Channels”](#) on page 417.

- 4** Associate the External Subscriber object you created in [Step 2](#) with the Channel you created in [step Step 3](#).

Other Subscribers from TREE1 can already be associated with this Channel.

For steps in associating Subscribers with Channels, see [“Associating Subscribers with Channels”](#) on page 424.

- 5** In TREE1, create the Distribution.

For steps in creating Distributions, see [“Distributions”](#) on page 393.

- 6** Associate this Distribution with the Channel you created in [Step 3](#).

- 7** Verify that the External Subscriber server in TREE2 received the Distribution.

TED Issues

- ◆ [“Understanding Dependencies in TED”](#) on page 443
- ◆ [“System Resources and Server Behavior”](#) on page 443

- ◆ “Controlling I/O Rates and Concurrent Distributions” on page 444
- ◆ “Minimizing Messaging Traffic” on page 445
- ◆ “Changing DNS Names or IP Addresses for TED Servers” on page 446
- ◆ “When a TED Process Fails” on page 446

Understanding Dependencies in TED

Policy and Distribution Services agents (Policy/Package Agent, Distributor Agent, and Subscriber Agent) are dependent on one another and upon eDirectory. It is important to understand the following dependencies when using Policy and Distribution Services to manage your network:

- ◆ “Synchronization of TED Objects in eDirectory” on page 443
- ◆ “Unloading Parent Subscribers” on page 443

Synchronization of TED Objects in eDirectory

ZfS uses eDirectory as the repository for information needed by the TED and Server Policies components. Since eDirectory is a distributed database and can have partitions and replicas throughout the network, it takes time to synchronize all of the replicas each time ZfS objects are created or modified.

The Distributor Agent and Policy/Package Agent are the only ones that read eDirectory. The Subscriber Agent does not.

Unloading Parent Subscribers

You must change the parent Subscriber attribute in the Subscriber object to change the parent Subscriber. Then, the next time a Distribution is sent, the distribution route to the Subscriber will reflect the new parent Subscriber.

If a parent Subscriber Java process is unloaded (exited), the subordinates of the parent Subscriber will not renegotiate to another parent Subscriber. The subordinates will wait until that parent Subscriber is loaded again and continue to use it. The reason for this is that if the parent Subscriber was the only server between twenty Subscribers and the Distributor (which is located across the WAN), you would not want all of the Subscribers to go across the WAN to get their Distributions if the parent Subscriber is unavailable.

System Resources and Server Behavior

Using Policy and Distribution Services can affect the behavior of your system:

- ◆ TED usage can affect system behavior because of the traffic created in sending Distributions
- ◆ Some server policies are designed to control the behavior of servers, such as how a server should be brought down
- ◆ Some server policies are designed for NetWare server configuration, such as SET parameters, content of the AUTOEXEC.NCF file, and so on

Installing and using TED can affect any of the following:

- ◆ CPU utilization
- ◆ Disk space resources
- ◆ Network traffic

- ◆ Other I/O activity

To optimize your installation of TED, you should consider the following issues when selecting Distributor and Subscriber servers:

- ◆ Which servers are the best candidates for the heavy workload of a Distributor?
Consider CPU speed for building and sending Distributions, and sufficient disk space for storing all of the Distributor's Distributions.
The server can perform other non-ZfS network functions, be running other ZfS or non-ZfS software, or it can be solely dedicated to the ZfS Distributor function.
- ◆ Which servers do you want to manage using server policies?
Consider installing the Subscriber software to each server that you want to manage with policies, or where you want to distribute software packages. The policy engine is installed with the Subscriber software; also, the Subscriber software is used to extract and install software packages.
- ◆ Which servers could best handle the additional workload of being a parent Subscriber? (A parent Subscriber is a Subscriber that acts as a proxy for the Distributor to store and pass Distributions so that the Distributor does not have to send its Distributions to every Subscriber.)
Consider CPU speed for sending the Distributions, and free disk space for storing the Distributions that the parent Subscriber will pass on.
- ◆ Does each of your LAN segments have servers that are capable of being a parent Subscriber?
Consider WAN traffic when deciding where to locate parent Subscribers.
- ◆ Do you have other processes using up bandwidth on some LANs and WAN links?
Consider Distribution priorities and setting sending and receiving rates to minimize the affect Distributions can have on bandwidth for WAN links.

Controlling I/O Rates and Concurrent Distributions

If you need to control bandwidth usage for Distribution traffic, you can set the I/O rates and the maximum number of concurrent Distributions for Distributors and/or Subscribers.

Attributes of both the Distributor and Subscriber objects provide the following controls:

- ◆ **Input Rate:** For sending and receiving Distributions, you can set the maximum bytes per second. The Distributor Agent and Subscriber Agent send and receive the Distributions. This allows you to have some control over the bandwidth used by these agents. The default is the maximum that the connection can handle. However, this does not control the rate at which FTP, HTTP, and RPM Distributions are built by the Distributor.
- ◆ **Output Rates Based Upon Distribution's Priority:** Sets the default output rate to minimize network traffic for TED objects. This determines the send rate for Subscribers. The default value is the maximum that the connection can handle. There are three output priorities where you can specify a rate:
 - ◆ **High Priority:** These Distributions will be sent before any Medium or Low priority Distributions.
 - ◆ **Medium Priority:** These Distributions will be sent after all High priority and before any Low priority Distributions.

- ◆ **Low Priority:** These Distributions will be sent after all High and Medium priority Distributions.

For more information, see [“Prioritizing Distributions” on page 410](#).

- ◆ **Maximum Number of Concurrent Distributions:** This determines how many simultaneous Distributions the Distributor Agent or Subscriber Agent will send. The default is unlimited (blank field). The Subscriber will always receive as many Distributions as it is sent; however, it will only concurrently pass on the number that you choose here.

If there is only one Subscriber, the Distributor will send Distributions at the selected rate. If there are two Subscribers, the Distributions will be sent at one half the rate. In other words, to determine the slowest distribution rate, divide the Distributor’s output rate by the maximum number of concurrent Distributions.

Because Subscribers will always receive another concurrent Distribution, the rate will still apply even though you cannot limit the number of incoming connections.

Minimizing Messaging Traffic

TED provides message notifications so that administrators and selected end users can be kept informed. Notifications can be sent in several ways:

- ◆ Information can be sent to log files
- ◆ Notifications can be sent via e-mail messages
- ◆ SNMP traps can be used and displayed on both local and remote consoles

The following sections explain notification usage:

- ◆ [“Message Notification Levels” on page 445](#)
- ◆ [“Sending Notifications Over LANs and WANs” on page 445](#)

Message Notification Levels

There are seven levels of messaging available, from no messages to be broadcast to a developer trace option. Regardless of the destination for a message, resources are directly affected by the level you choose. For information on setting message levels, see:

- ◆ **Distributor object:** [Step 3 on page 390](#)
- ◆ **Subscriber object:** [Step 3 on page 421](#)

The level you choose for a log file will affect the rate at which the log file grows. Because log files have no maximum size, you can control the size of a log file by choosing to delete entries after x number of days. For information on setting message levels, see:

- ◆ **Distributor object:** [Step 2 on page 389](#)
- ◆ **Subscriber object:** [Step 2 on page 420](#)

Sending Notifications Over LANs and WANs

The greatest impact on network traffic can come from the levels you choose for SNMP traps and for the remote console.

For information on setting message levels for SNMP traps, e-mail messages, and the server’s console, see:

- ◆ **Distributor object:** [Step 3 on page 390](#)
- ◆ **Subscriber object:** [Step 3 on page 421](#)

SNMP Traps

SNMP messages are sent only if there is an SNMP policy in effect for the receiving server, regardless of the level you choose for the messages. SNMP traffic is affected by both the level you choose and by the SNMP configuration in the policy on the server. There is one SNMP packet per message per destination in the SNMP Trap Target policy. IPX™ addresses are not supported for trap targets.

E-Mail Messages

E-mail messages can also affect network traffic. Like SNMP, e-mail will send only one e-mail per message per e-mail user defined. E-mail is also configured by a server policy. You must define and enable the policy on the sending server for e-mail messages to be sent.

Changing DNS Names or IP Addresses for TED Servers

Whenever there is a change to the identity of either a Distributor or Subscriber server, you must perform certain tasks so that the distribution processes for these servers can continue as before.

In the distribution process, TED servers identify themselves to each other by their DNS names or IP addresses. The following sections explain situations that can arise from changing these server identifiers.

If You Are Using DNS Names to Identify Your Servers

- ◆ If you change the DNS name of a Distributor server, Subscriber servers will no longer be able to recognize the Distributor as a valid source for receiving Distributions.
- ◆ If you change the DNS name of a Subscriber server, the Distributor will not be able to locate the Subscriber server for sending Distributions to it. This is because the Distributor obtains the Subscriber server's address from the eDirectory object.

If you change the IP address of a Distributor or Subscriber server when you are using its DNS name to identify it to ZfS, this change will not affect the distribution processes.

If You Are Using IP Addresses to Identify Your Servers

- ◆ If you change the IP address of a Distributor server, Subscriber servers will no longer be able to recognize the Distributor as a valid source for receiving Distributions.
- ◆ If you change the IP address of a Subscriber server, the Distributor will not be able to locate the Subscriber server for sending Distributions to it. This is because the Distributor obtains the Subscriber server's address from the eDirectory object.

Because reinstating valid certificates is involved in resolving server identity changes, see [“Handling Invalid Certificates” on page 553](#) for instructions.

When a TED Process Fails

It is possible, for many common computer-related reasons, that a TED process could fail. The following are a few possibilities:

- ◆ **A Distribution could be interrupted.** If so, when it restarts it will pick up where it left off.

Before distribution, the Distribution package resides at the Distributor. After distribution, the Distribution package still resides at the Distributor with a copy now at the Subscriber. It is during the distribution process that an interruption could halt copying. When the Distributor tries to re-send the Distribution (the next time the Channel schedule starts), it will pick up where it left off and not re-send the entire Distribution.

If the re-sending of a Distribution is interrupted, the sender will retry every two minutes for 30 minutes. If it is not successful in reestablishing connection to the target server, it will stop retrying. The next time the Channel's schedule starts it will pick up where it left off in sending the Distribution when it was originally interrupted.

- ◆ **An extraction could be interrupted.** If so, the extraction will not pick up where it left off.

Distributions are made across the wire from server to server, while extractions are performed on the server from Distributions already sent. Therefore, when an extraction is interrupted, it simply fails. The Subscriber will not roll back (or undo) the failed extraction, unless the Distribution was a software package (.CPK file). It will try the extraction again the next time the Subscriber's extraction schedule starts.

Files are extracted to the volume and directory specified when the Distribution package was created. File groupings and software packages both allow you to specify which volume and directory the package should be extracted to. Therefore, when an interruption occurs during extraction, it fails in the same way as if you were copying a file in the operating system.

- ◆ **The File type offers the following:**

- Retry *X* times
- Kill the connection on files that are open
- Error handling (Fail on error; perform a routine on error)

All options deal with extraction and how to handle it.

Working Directories

Distributors and Subscribers use working directories on the servers for Distributions, patches, status files, and temporary working files. The size of a working directory is determined by the size and number of Distributions.

The working directories default to the SYS: volume on NetWare servers or the C: Drive on Windows servers. Because of disk space considerations on NetWare servers, we recommend that you select a different location on the server, such as a DATA: volume.

The default working directory names for NetWare and Windows servers are *Path\ZENWORKS\PDS\TED\DIST* for the Distributor and *Path\ZENWORKS\PDS\TED\SUB* for the Subscriber. For Linux and Solaris servers, the paths are *usr/ZENworks/PDS/TED/Working/Dist* and *usr/ZENworks/PDS/TED/Working/Sub*. You can change working directory names in the properties of the TED object.

The following sections describe the TED directory structures:

- ◆ “NetWare Distributor Directories” on page 448
- ◆ “NetWare Subscriber Directories” on page 449
- ◆ “Windows NT Distributor Directories” on page 450
- ◆ “Windows NT Subscriber Directories” on page 450
- ◆ “UNIX Distributor Directories” on page 450

- ◆ “UNIX Subscriber Directories” on page 451

NetWare Distributor Directories

The following directories are used by NetWare Distributors:

volume:\installation_path\ZENWORKS\PDS\TED

Contains the TED software for the Distributor.

volume:\installation_path\ZENWORKS\PDS\TED\Security\Private

Contains the Distributor’s private key.

volume:\working_directory

Contains one subdirectory for each Distribution that belongs to the Distributor. The working directory name is user-defined in the Distributor object.

volume:\working_directory\Distribution_directory

Each Distribution has its own subdirectory that is created under the working directory. The Distribution directory’s name is derived from the following syntax: *Tree_DN_of_Distribution*. For example, TestTree_Files.TED.Novell.

volume:\working_directory\Distribution_directory\time_stamp_directory

Each Distribution directory contains multiple time stamp directories, which are named according to the date and time the Distribution was built.

Each time a Distribution is built, the Distributor checks to see if anything has changed since the last time the Distribution was built. If so, a new time stamp directory is created.

The number of time stamp directories kept is determined by the Maximum Number of Revisions to Keep field in the Distribution object’s properties. There are occasions when the number of time stamp directories will exceed the maximum number specified because the Distributor will not delete a time stamp directory that is in use. The Distributor removes the oldest time stamp directories first.

Sometimes a time stamp directory name will have *_TEMP* appended to it. When a Distributor builds a Distribution, it creates a **_TEMP* directory before it determines if anything has changed. If changes are discovered, the *_TEMP* is removed and the directory is used for the new build.

A Distributor’s time stamp directories contain the following files:

Filename	Description
DISTFILE.TED	The Distribution that was built. All Distributions have the same filename. They are distinguished by their time stamp directory’s name and path.

Filename	Description
<i>digest_file</i>	<p>This file will only exist if the Distributor Agent creates it (optional).</p> <p>Digests are used by Distributors and Subscribers to verify that Distributions have not been tampered with while in transit. The digest provides a checksum for the Subscriber to compare.</p> <p>The syntax for creating the digest filename is:</p> <p style="padding-left: 40px;">%AGENT%AgentDigest.TED</p> <p>For example:</p> <p style="padding-left: 40px;">FTPAgentDigest.TED</p> <p style="padding-left: 40px;">HTTPAgentDigest.TED</p> <p style="padding-left: 40px;">FileAgentDigest.TED</p> <p style="padding-left: 40px;">CPKAgentDigest.TED</p>

NetWare Subscriber Directories

The following directories are used by NetWare Subscribers:

volume:\installation_path\ZENWORKS\IPDS\TED

Contains the TED software for the Subscriber and/or Distributor.

volume:\installation_path\ZENWORKS\IPDS\TED\Security

Contains certificates received from Distributors.

volume:\working_directory

Contains one subdirectory for each Distribution that it receives from a Distributor. The working directory name is user-defined in the Subscriber object.

volume:\working_directory\Distribution_directory

Each Distribution has its own subdirectory that is created under the working directory. The Distribution directory's name is derived from the following syntax: *Tree_DN_of_Distribution*. For example, TestTree_Files.TED.Novell.

volume:\working_directory\Distribution_directory\time_stamp_directory

Each Distribution directory contains multiple time stamp directories, which are named according to the date and time the Distribution was built.

The number of time stamp directories kept is determined by the Maximum Number of Revisions to Keep field in the Distribution object's properties.

Once a threshold is met, the Subscriber receives the maximum revision information and deletes the oldest time stamp directories first.

A Subscriber's time stamp directories contain the following files:

Filename	Description
DISTFILE.TED	The Distribution that was built. All Distributions have the same filename. They are distinguished by their time stamp directory's name and path.
DISTSTATUS.TED	Once a Distribution has been successfully received, this file is created.
<i>digest_file</i>	This file will only exist if the Distributor Agent has created it (optional). Digests are used by Distributors and Subscribers to verify that Distributions have not been tampered with while in transit. The digest provides a checksum for the Subscriber to compare.

Windows NT Distributor Directories

The following directories are used by Windows NT Distributors:

installation_path\ZENWORKS\PDSTED

Contains the TED software for the Distributor.

installation_path\ZENWORKS\PDSTED\Security\Private

Contains the Distributor's private key.

Windows NT Subscriber Directories

The following directories are used by Windows NT Subscribers:

installation_path\ZENWORKS\PDS

Contains the TED software for the Subscriber.

installation_path\ZENWORKS\PDSTED\Security\Private

Contains certificates received from Distributors.

local_drive:\working_directory\Distribution_directory\time_stamp_directory

Each Distribution directory contains multiple time stamp directories, which are named according to the date and time the Distribution was built.

UNIX Distributor Directories

The following directories are used by UNIX Distributors:

usr/ZENworks/PDS/TED/Working/Dist

Contains the TED software for the Distributor.

usr/ZENworks/PDS/TED/Security/Private

Contains the Distributor's private key.

Each Distribution directory contains multiple time stamp directories, which are named according to the date and time the Distribution was built.

UNIX Subscriber Directories

The following directories are used by UNIX Subscribers:

usr/ZENworks/PDS/TED/Working/Sub

Contains the TED software for the Subscriber.

usr/ZENworks/PDS/TED/Security/Private

Contains certificates received from Distributors.

Each Distribution directory contains multiple time stamp directories, which are named according to the date and time the Distribution was built.

Editing the TEDNODE.PROPERTIES File

If you should install the Subscriber software to a server that will not have a Subscriber object in any eDirectory tree, such as a Windows server in a Microsoft domain, the TEDNODE.PROPERTIES file will be used by such a server for its configuration information. For configuration changes, you would need to edit the server's TEDNODE.PROPERTIES file using the information in this section.

The TEDNODE.PROPERTIES file is located in the ZENWORKS\PDSTED directory on the server.

Following is the required format of the file, including comments on some of the entries. Note that the information on the right side of an = symbol is only an example and not the required value for that line. However, the examples are intended to show the correct syntax for the values.

Line Content	Comments
workingdir = d:\ted\tran	Subscriber's working directory
io.input = 100	Receive rate in bytes per second
io.output = -1	Send rate in bytes per second
variable1 = vol=sys:	Define the variable "vol" with the value "sys:"
variable1.description = Destination Volume	A description of the variable's function
console.level = 6	Message level for the server's console
log.level = 1	Message level for log file
log.days = 1	Number of days to save log file entries
log.path = d:\ted\tran\log.txt	Path for log file and log filename
workorder.timeout = 0	Number of seconds to wait for reply from the Distributor before dropping connection; 0 = wait forever

Line Content	Comments
workorder.concurrent = 0	Concurrent Distributions
email.level = 0	Message level for e-mail
smtp.host = email.novell.com	Location of SMTP host
snmp.level = 0	Message level for SNMP traps
email.target1 = johndoe@novell.com	E-mail address for the messages

For the remaining TEDNODE.PROPERTIES file entries, remove the # symbol from a line to enable it. This will make that line effective for the schedule type if is listed under. However, do not remove the # symbol from the first line for a schedule type because it is only a description that indicates the schedule type. You can change the default values that are listed.

Note that the following sample has the Daily schedule enabled because the appropriate # symbols have been removed.

Line Content
Yearly schedule and associated keys (with default values specified)
#schedule.type=yearly
#schedule.month=1
#schedule.day=1
#schedule.begin.hour=8
#schedule.begin.minute=0
#schedule.end.hour=17
#schedule.end.minute=0
#schedule.random=false
Monthly schedule and associated keys (with default values specified)
#schedule.type=monthly
#schedule.day=1
#schedule.begin.hour=8
#schedule.begin.minutes=0
#schedule.end.hour=17
#schedule.end.minute=0
#schedule.random=false
Daily schedule and associated keys (with default values specified)
schedule.type=daily
schedule.days=Monday,Tuesday,Wednesday,Thursday,Friday,Saturday,Sunday
schedule.begin.hour=8
schedule.begin.minutes=0
schedule.end.hour=17
schedule.end.minute=0
schedule.repeat.days=0
schedule.repeat.hours=0
schedule.repeat.minutes=0
schedule.random=false
Immediate schedule and associated keys (with default values specified)
#schedule.type=immediately
#schedule.repeat.days=0
#schedule.repeat.hours=0
#schedule.repeat.minutes=0

Line Content

Interval schedule and associated keys (with default values specified)

#schedule.type=interval

#schedule.repeat.hours=0

#schedule.repeat.minutes=0

Never schedule and associated keys (with default values specified)

#schedule.type=never

Time schedule and associated keys (with default values specified)

#schedule.type=time

#schedule.date.year=2001

#schedule.date.month=1

#schedule.date.day=1

#schedule.begin.hour=8

#schedule.begin.minutes=0

17

Server Policies

Novell® ZENworks® for Servers (ZfS) provides server policies for managing server configurations, processes, and behaviors.

The following sections will help you to understand, set up, and configure the policies:

- ◆ [“Understanding Server Policies” on page 455](#)
- ◆ [“Creating a Policy Package” on page 465](#)
- ◆ [“Configuring Server Policies” on page 467](#)
- ◆ [“Enabling Policies” on page 487](#)
- ◆ [“Distributing Policies” on page 487](#)
- ◆ [“Associating Policies” on page 487](#)
- ◆ [“Scheduling Policies” on page 488](#)
- ◆ [“Viewing Effective Policies” on page 489](#)
- ◆ [“Changing Policy Enforcement” on page 490](#)

Understanding Server Policies

In ZfS 3.0.2, most policies are enforced through the distribution of policy packages. However, a few policies used by the Distributor are enforced by being associated with Novell eDirectory™ containers. Prior to ZfS 3.0.2, all policies were enforced through container and object associations.

Review the following sections to understand policies in ZfS 3.0.2:

- ◆ [“Configuration and Behavioral Management through Server Policies” on page 455](#)
- ◆ [“Server Policies and Packages” on page 456](#)
- ◆ [“Plural and Cumulative Policies” on page 456](#)
- ◆ [“Server Policies Architecture” on page 457](#)
- ◆ [“Configuration and Behavioral Policies” on page 457](#)
- ◆ [“Enforcing Policies” on page 460](#)
- ◆ [“Server Policy Descriptions” on page 461](#)

Configuration and Behavioral Management through Server Policies

The Server Policies component provides configuration and behavioral management of your servers. Server policies are divided into three packages for the convenience of scheduling policies and distributing the policies to their applicable servers:

- ◆ **Container Package:** Holds the Search policy that determines how Policy and Distribution Services searches eDirectory for objects associated with policies.
- ◆ **Server Package:** Has a generic set of policies that can be applied to all servers, as well as policy package sets for servers on specific platforms. This package is provided for backwards compatibility with ZfS 2 and for certain components that require policies to be associated for enforcement.
- ◆ **Distributed Server Package:** Has a generic set of policies that can be applied to all servers, as well as policy package sets for servers on specific platforms. This package is new for ZfS 3.0.2 and provides policies that are distributed for enforcement.
- ◆ **Service Location Package:** Holds policies specific to running Policy and Distribution Services.

Configuration policies hold information in eDirectory that creates a similar type of configuration on a server, such as enforcing selected SET parameters. Behavioral policies hold a set of rules to be followed under certain situations, such as when a server goes down.

Through server policies you can automate the management of your servers, and through ConsoleOne® and the ZfS Management role in Novell iManager you can configure policies and manage your servers from a single workstation.

Server Policies and Packages

Server policies provide you with the ability to set, standardize, and automate configuration parameters on any given set of servers. You can control the behavior of servers in given situations, such as downing a server.

To use server policies, you must first create the appropriate Policy Package objects in ConsoleOne, configure the policies you need, enable them, and distribute the package to the applicable Subscriber servers where the package's policies are enforced.

When you set up server policies, you can individually schedule them to run daily, weekly, monthly, yearly, by an event, at a specific date and time, relative to a date and time, by an interval of time, or even immediately. The schedule individual policies use is the default for their policy package's schedule, which you can change.

Any or all of the Policy and Distribution Services policies can be implemented in a policy package. You can also create a Policy Package object for each different configuration set that you need. For example, you could want some of your servers to be brought down differently.

All policies enabled in a package will be enforced on any servers where the Policy Package type Distribution has been received and extracted.

Plural and Cumulative Policies

Policy packages can contain both plural and cumulative policies. All plural policies are also cumulative, but cumulative policies are not necessarily plural. For more detail, review:

- ◆ [“Plural Policies” on page 456](#)
- ◆ [“Cumulative Policies” on page 457](#)

Plural Policies

Plural policies are those where there can be more than one per policy package per platform.

For example, in the same policy package, you can add and configure a Scheduled Down policy and name it "Scheduled Down for Time A." Then you could add and configure another Scheduled Down policy, this time naming it "Scheduled Down for Time B."

You can tell if a policy is plural by viewing the Policies tab and clicking Add, because all plural policies are listed in the Add dialog box.

Cumulative Policies

Cumulative policies are those that allow multiples of the same policy to be in effect when multiple policy packages are distributed to a server. For example, a Text File Changes policy distributed to Server A could be accumulated with a differently configured Text File Changes policy distributed to Server A. All of the text file changes from both policies would be effective for Server A.

Configuration and Behavioral Policies

A single configuration policy can affect the configuration of a single server or many servers. For example, a policy can be scheduled to run at regular intervals to ensure that the server's configuration continues to be set correctly.

Behavioral policies hold a set of rules to be followed in certain situations. The policy engine carries out these rules, along with any of its supporting modules. For example, the Server Down Process policy defines criteria that must be met before the server can be brought down, such as:

- ◆ How soon before the server is brought down should users be notified
- ◆ Who is notified when the policy is being enforced
- ◆ Which peer server will send SNMP alerts if the server does not come back up

Behavioral policies are designed to make servers act more intelligently, to handle situations an administrator might not even be aware of, and to reduce complexity for administrators.

In summary, the benefits of configuration and behavioral policies include:

- ◆ Automating tasks that an administrator would normally perform
- ◆ Notifying specified users through e-mail messages that a server is going down
- ◆ Allowing a server down process to abort on certain conditions

Server Policies Architecture

To understand how server policies are used to manage your servers, you must understand its eDirectory objects and its agent:

- ◆ [“eDirectory Schema Extensions for Server Policies” on page 457](#)
- ◆ [“Policy/Package Agent” on page 460](#)

eDirectory Schema Extensions for Server Policies

The eDirectory schema extensions included in the Server Policies component define the class of eDirectory objects that can be created in your eDirectory tree, including which information is required or optional at the time the object is created. Every object associated with the Server Policies component in an eDirectory tree has a class defined for it in the tree's schema.

ZFS objects for the eDirectory schema are:

Container Package
Server Package
Service Location Package
Distributed Server Package
ZENworks Database

Note the following concerning policy enforcement:

- ◆ All of the policies in the Distributed Server Package must be distributed to be enforced (ZfS 3.0.2 servers only)
- ◆ All of the policies in the Container Package, Server Package, and Service Location Package must be associated to be enforced (ZfS 2 and ZfS 3.0.2 servers)

The Server Package provides backwards compatibility that allows you to run ZfS 3.0.2 and ZfS 2 concurrently, such as during upgrading.

Existing eDirectory classes that are modified with the addition of ZfS attributes are:

Country
Group
Locality
Organization
Organizational Unit
Server

The following sections summarize the primary eDirectory objects that are added to eDirectory from the schema extensions provided with the Server Policies component:

- ◆ “Container Package Object” on page 458
- ◆ “Server Package Object” on page 458
- ◆ “Service Location Package Object” on page 459
- ◆ “Distributed Server Package” on page 459
- ◆ “ZENworks Database Object” on page 459

For basic information about the types of objects in an eDirectory tree, see the [Novell Documentation Web site \(http://www.novell.com/documentation/lg/nw5/docui/index.html\)](http://www.novell.com/documentation/lg/nw5/docui/index.html) and select Procedures > Planning > Directory Services > eDirectory Planning.

Container Package Object

The Container Package object is an eDirectory object that manages the Search policy object. This policy is used by the Distributor and Subscriber objects for all versions of ZfS, and must be associated to be enforced.

Server Package Object

The Server Package object is an eDirectory object that manages the following policy objects for ZfS 2 backwards compatibility and one policy for ZfS 3.0.2 Server Inventory:

Copy Files (ZfS 3.0.2 only)
NetWare Set Parameters
Scheduled Down
Scheduled Load/Unload
Server Down Process
Server Scripts

- SNMP Community Strings
- SNMP Trap Target Refresh (ZfS 2 only)
- Text File Changes
- ZENworks Database (ZfS 3.0.2 Server Inventory only)
- ZENworks for Servers

Server Package policies are used for configuring servers and controlling server behavior.

All policies in this package must be associated to be enforced.

Service Location Package Object

The Service Location Package object is an eDirectory container object that manages the following policy objects:

- SMTP Host
- SNMP Trap Targets
- Tiered Electronic Distribution
- ZENworks Database
- ZENworks for Servers License (ZfS 2 only)

Service Location Package policies provide general Policy and Distribution Services configuration and location information.

All policies in this package must be associated to be enforced.

All policies except ZENworks for Servers License are used by ZfS 3.0.2 Distributors and Subscribers.

Distributed Server Package

The Distributed Server Package object is an eDirectory object that manages the following policy objects (ZfS 3.0.2 only):

- Copy Files
- NetWare Set Parameters
- Scheduled Down
- Scheduled Load/Unload
- Server Down Process
- Server Scripts
- SMTP Host
- SNMP Community Strings
- SNMP Trap Targets
- Text File Changes
- ZENworks Database
- ZENworks for Servers

Distributed Server Package policies are used for configuring servers, controlling server behavior, and providing general ZfS configuration and location information.

All policies in this package must be distributed to be enforced.

ZENworks Database Object

Provides the location of the ZFSLOG.DB file for logging reporting information. The database file can be installed on NetWare[®] and Windows servers.

The ZENworks Database object can exist multiple times in a tree, each with its own associated database file; however, there can only be one database file installed per server.

The Server Policies component writes policy information to the ZENworks database (ZFSLOG.DB). Because every server in your network can be running the Policy/Package Agent, they can each write to the database, even across WAN links. If you do not need consolidated server policies reports on all servers, you can install a database to each WAN segment.

If you require consolidated server policies reports, you can have just one ZFSLOG.DB file where all servers running the Policy/Package Agent will log information. The amount of data a Policy/Package Agent writes to the database might not create excessive WAN traffic, depending on the number of servers and speeds of the WAN links.

Because you can install the ZENworks database to multiple servers, to minimize WAN traffic you should coordinate the placement of Policy Package and ZENworks Database objects in containers on the WAN segments.

Policy/Package Agent

Policy and Distribution Services allows you to manage your network servers using the Policy/Package Agent. This agent is installed on each server where you select the Subscriber/Policies installation option.

The Policy/Package Agent does the following:

- ◆ Extracts (installs) a software package's contents.
- ◆ Extracts the policy information from a Policy Package Distribution.
- ◆ Enforces the enabled policies from the extracted policy information based on their enforcement schedules.

There are a number of server policies that provide configuration and behavioral management of your servers. The Policy/Package Agent must be running on each server you want to manage with policies or have software packages to extract and install.

The Policy/Package Agent should be installed to every server in your network. Exceptions might be servers where you do not need to distribute software packages, or servers that you do not want to manage using policies.

Enforcing Policies

Most ZfS 3.0.2 policies are enforced by creating the policy package, enabling and configuring the policy, scheduling the package, distributing the package, and extracting the policies on servers.

Some ZfS 3.0.2 policies are enforced by creating the policy package, enabling and configuring the policy, scheduling the package, and associating the package with the containers where the Distributor or Subscriber objects reside.

For more information, review the following:

- ◆ [“Scheduling Policies” on page 461](#)
- ◆ [“Distributing Policies” on page 461](#)
- ◆ [“Associating Policies” on page 461](#)

Scheduling Policies

Some server policies must be scheduled before they can be enforced.

The following schedules can be used:

- ◆ Activate by the Default Package Schedule (which can be set to any of the schedules)
- ◆ Activate on a specified event (such as running at system startup or shutdown)
- ◆ Activate once relative to a period of time
- ◆ Activate at a specified date and time
- ◆ Activate once per year at a specified time
- ◆ Activate once each month at a specified time
- ◆ Activate on one or more days of the week at specified times
- ◆ Activate on one or more days of the week, repeating at a specified interval of time
- ◆ Continuously repeat at a specified interval of time
- ◆ Run immediately
- ◆ Run immediately, repeating at a specified interval of time

IMPORTANT: If you enable a policy, but do not schedule it, it will activate according to the schedule currently specified in the Default Package Schedule.

The Default Package Schedule provides a default for unscheduled policies in the policy package. The default schedule is the Run At System Startup event.

Distributing Policies

Once you have enabled and configured a policy contained in the Distributed Server Package, you must distribute its policy package to the Subscriber servers where the enabled policies can be placed into effect. In other words, configuring and enabling a policy only sets up the policy. It is enforced through its distribution to and extraction on the applicable servers that are running Policy and Distribution Services.

Associating Policies

Once you have enabled and configured a policy contained in the Server Package or Service Location Package, you must associate its policy package with the containers where Distributor or Subscriber objects reside so that the enabled policies can be placed into effect. This association can be directly with a container where the Distributor or Subscriber objects reside, or with a container higher in the tree from where the container holding these objects reside.

Because configuring and enabling a policy only sets up the policy, it is enforced through its association with the applicable servers that are running Policy and Distribution Services.

Server Policy Descriptions

The following tables list the server policies by policy package. The second column indicates whether a policy is a configuration or behavioral policy, and whether it is cumulative, plural, or both.

- ◆ [“Container Package” on page 462](#)
- ◆ [“Service Location Package” on page 462](#)

- ◆ “Server Package” on page 463
- ◆ “Distributed Server Package” on page 464

Container Package

This policy description only applies to ZfS 3.0.2. See your ZfS 2 documentation for details on how the Search policy might be used differently for ZfS 2 servers running concurrently with ZfS 3.0.2.

Policy Name	Policy Type Keys	Policy Function
Search	Behavioral	<p>If you don't set a Search policy, the default is to search from the parent container to the root every hour. This can create unnecessary search traffic. Therefore, we recommend that you make effective use of the Search policy.</p> <p>This Search policy can only be administered in ConsoleOne. A Search policy created in NetWare Administrator for ZENworks will not be recognized in ZfS.</p>

Because most policies in ZfS are distributed rather than associated for enforcement and a Distributor does not receive Distributions, the Search policy is used in ZfS to enable the Distributor Agent to locate and use policies in the Service Location Package. For example, the Distributor Agent uses the package's ZENworks Database policy to write reporting information to the ZfS Database file.

Also, Distributors read the Service Location Package policies for their Subscribers. That means Subscribers receive their Service Location Package policies through associations, as well.

Service Location Package

This policy package is used by both ZfS 2 and ZfS 3.0.2.

Policy Name	Policy Type Keys	Policy Function
SMTP Host	Configuration	Sets the TCP/IP address of the relay host that processes outbound Internet e-mail. This policy must be enabled if you select the E-Mail option for notifying or logging messages in any of the other policies.
SNMP Trap Targets	Configuration	<p>Sets SNMP trap targets for associated eDirectory objects.</p> <p>In ZfS 3.0.2, this policy can be scheduled for when you want it to be refreshed. In ZfS 2, the SNMP Trap Targets Refresh policy contained in the Server Package must be used for scheduling this policy.</p> <p>IPX™ addresses are not supported for SNMP trap targets. Only IP addresses and DNS names can be used.</p>

Policy Name	Policy Type Keys	Policy Function
Tiered Electronic Distribution	Configuration	<p>Sets defaults for the Distributor and Subscriber objects, including:</p> <ul style="list-style-type: none"> I/O rates Maximum concurrent Distributions Connection time-out in minutes Working directory Parent Subscriber Messaging levels for a server's console, SNMP traps, log files, and e-mail notification Extraction Schedule Refresh Schedule Variables <p>Note that any defaults set here override unchanged defaults in a TED object. However, if a TED object's properties are modified, those modifications have precedence over any defaults set in the TED policy.</p>
ZENworks Database	Configuration	<p>Sets the DN for locating the ZENworks Database object. This policy must be in effect for Policy and Distribution Services to locate a database for logging successes and failures that are used in creating reports.</p> <p>If a database object is not identified with this policy, Policy and Distribution Services will not use the database to log reporting information. Therefore, you should create this policy to identify the database.</p> <p>The Policy/Package Agent and the Distributor Agent both write to ZFSLOG.DB. For information on having these agents write to different database files, see "Coexisting Databases" on page 589.</p>
ZENworks for Servers License	Configuration	<p>ZfS 2 only. Identifies the NLS object, otherwise ZfS 2 Policy and Distribution Services will not work.</p>

Server Package

The Server Package exists in ZfS 3.0.2 for backwards compatibility with ZfS 2, such as when upgrading incrementally. This package also exists to provide policies that must be associated, such as for ZfS 3.0.2 Server Inventory or ZENworks for Desktops (ZfD) 3.x or 4.0.1. ZfD would add its own policies to this package when installed.

From a ZfS perspective, this package can display different policies, depending on whether ZfS 2 and ZfS 3.0.2 exist in a mixed environment. For example:

- ◆ The ZENworks Database policy did not exist in ZfS 2, yet it is displayed in this package. Only ZfS 3.0.2 Server Inventory can use this policy.
- ◆ The Copy Files policy did not exist in ZfS 2, yet it is displayed in this package as a policy that can be added. Only ZfS 3.0.2 servers can use this policy.
- ◆ The SNMP Trap Target Refresh policy will not display if only ZfS 3.0.2 is installed. If the ZfS 2 snap-ins are also present, this policy will then be displayed. Only ZfS 2 servers can use this policy.

There are several policies that are used in ZfS 2 that the ZfS 3.0.2 version of the package will not display, unless the ZfS 2 snap-ins are also present.

In order to manage ZfS 2 servers using the ZfS 3.0.2 Server Package, you must have done the following during upgrading:

1. Updated the ConsoleOne version that ZfS 2 is using by installing version 1.3.5 over it from the *ZENworks for Servers Companion CD* or *ZENworks 6 Companion 1 CD*.
2. Installed the ZfS 3.0.2 snap-ins to the updated version of ConsoleOne.

After you have done this, you will be able to manage your ZfS 2 servers using the ZfS 3.0.2 version of the Server Package. You will not need to re-create any Server Packages that you created in ZfS 2, because by installing ZfS 3.0.2 snap-ins to the same instance of ConsoleOne where the ZfS 2 snap-ins reside, the existing Server Packages are effectively updated for management using ZfS 3.0.2.

Although the ZENworks Database policy did not exist in ZfS 2, it will be displayed in this package. Only the ZfS 3.0.2 Server Inventory component uses the ZENworks Database policy. For more information, see [“Configuring the Database Location Policy” on page 710](#).

The following table only lists the ZENworks Database policy. For information on the other policies in the Server Package, see the ZfS 2 documentation on the [Novell Documentation Web site \(http://www.novell.com/documentation/lg/zfs2/index.html\)](http://www.novell.com/documentation/lg/zfs2/index.html).

Policy Name	Policy Type Keys	Policy Function
ZENworks Database	Configuration	Sets the DN for locating the ZENworks Database object. This policy must be in effect for Server Inventory to locate a database for logging inventory data.

Distributed Server Package

This package contains the policies the must be distributed to ZfS 3.0.2 servers to be enforced on them.

Policy Name	Policy Type Keys	Policy Function
Copy Files	Plural Cumulative Configuration	Enables copying of files on a server from one location to another by using policy configurations.
NetWare Set Parameters	Plural Cumulative Configuration	Specifies and optimizes selected Set Parameters for a server or group of servers. For the NetWare platform only.
Scheduled Down	Plural Cumulative Configuration Behavioral	Schedules when a server should go down, and whether it should be automatically brought back up. The policy includes which command to use in bringing it down (RESET, RESTART, or DOWN).
Scheduled Load/Unload	Plural Cumulative Configuration	For automating the loading and unloading order of NLM™ and Java Class processes for the selected servers, and for starting and stopping Windows services. NLM files that require user input to unload cannot be automated.
Server Down Process	Behavioral	For controlling which processes to follow and which conditions to meet before downing a server.

Policy Name	Policy Type Keys	Policy Function
Server Scripts	Plural Cumulative Configuration	For automating script usage on your servers.
SMTP Host	Configuration	Sets the TCP/IP address of the relay host that processes outbound Internet e-mail. This policy must be enabled if you select the E-Mail option for notifying or logging messages in any of the other policies.
SNMP Community Strings	Configuration	Allows you to receive and respond to SNMP requests.
SNMP Trap Targets	Configuration	Sets SNMP trap targets for associated eDirectory objects. This policy can be scheduled for when you want it to be refreshed. IPX addresses are not supported for SNMP trap targets. Only IP addresses and DNS names can be used.
Text File Changes	Plural Cumulative Configuration	For automating changes to text files.
ZENworks Database	Configuration	Sets the DN for locating the ZENworks Database object. This policy must be in effect for Policy and Distribution Services to locate a database for logging successes and failures that are used in creating reports. If a database object is not identified with this policy, Policy and Distribution Services will not use the database to log reporting information. Therefore, you should create this policy to identify the database. The Policy/Package Agent and the Distributor Agent both write to ZFSLOG.DB. For information on having these agents write to different database files, see “Coexisting Databases” on page 589 .
ZENworks for Servers	Configuration	Basic configuration parameters for Policy and Distribution Services, such as status logging, defining the server console prompt for the Policy/Package Agent, setting its working path, and setting a database purging limit. This policy can be enabled on each server where you want to enforce server policies. However, if you do not enable the policy, Policy and Distribution Services will work from pre-programmed defaults.

Creating a Policy Package

Policy and Distribution Services groups its server policies into four Policy Package objects:

- ◆ Container Package
- ◆ Server Package (ZfS 2 compatibility)
- ◆ Service Location Package
- ◆ Distributed Server Package (ZfS 3.0.2 only)

You can place policy packages anywhere in the tree. For ease of management, we recommend that you create an OU container for grouping the policy packages. For example, Policies.

However, if you install ZENworks for Desktops (ZfD) to your tree, you could keep the ZfS and ZfD policies in separate containers, such as ZfS_Policies and ZfD_Policies.

IMPORTANT: If you have partitions that are accessed across a WAN, make sure that the Policy Package objects are in the same partition as the Server object to ensure that the Policy/Package Agent will load. Also make sure that the Search policy does not require searching outside the partition where the Server object exists.

To determine which Policy Package objects to create, first determine which policies you will need.

To create Policy Package objects, review the instructions in the following sections:

- ◆ “Creating a Policies Container” on page 466
- ◆ “Creating a Policy Package Object” on page 466

Creating a Policies Container

To create the OU container object for holding your Policy Package objects:

- 1** In ConsoleOne, right-click the container where you want the policies container located.

IMPORTANT: Where you create the OU, and how many characters you use to name it, will directly affect the number of characters that you will have available for naming the plural policies. eDirectory has a 64-character limit for the full name and path in the tree for a policy.

Because you can have many different versions of one plural policy in a single policy package, you will want to be able to name them descriptively. Therefore, place the OU as high in the tree as is logical, and give it a short name to provide as many characters as possible for naming the policies.

- 2** Click New > Object > Organizational Unit.
- 3** Name the OU > click OK.

Creating a Policy Package Object

To create a Policy Package object:

- 1** In ConsoleOne, right-click the container you created for the Policy Package objects > click New > click Policy Package.

The Policy Package Wizard opens.

- 2** Under Policy Packages, select a policy package > click Next.

Available packages include: Container, Server, Service Location, and Distributed Server.

- 3** Name the package > click Next.

Because you can have multiples of the same package type, use a unique, informative name for each package.

IMPORTANT: Because of the eDirectory 64-character path/name limit, and the package name you enter here will be part of the path for plural policies that you can create later, enter a brief, but unique, Policy Package object name so that you will have as many characters as possible to be available for giving descriptive plural policy names.

- 4** Repeat **Step 2** and **Step 3** for each package to be created.

Click the Create Another Policy Package check box to save repeating **Step 1**.

Configuring Server Policies

You can configure server policies for containers, servers, and service locations. The policies allow you to automate use of NetWare functionality. See your NetWare documentation for specific information.

To configure server policies, review the instructions in the following sections:

- ◆ “[Compiling ZENTRAP.MIB](#)” on page 467
- ◆ “[Configuring the Container Package Policy](#)” on page 467
- ◆ “[Configuring Server Package Policies](#)” on page 469
- ◆ “[Configuring Service Location Package Policies](#)” on page 469
- ◆ “[Configuring Distributed Server Package Policies](#)” on page 475
- ◆ “[Creating Custom Log Files Using Policies](#)” on page 486

For information on scheduling server policies, see “[Scheduling Policies](#)” on page 488.

Compiling ZENTRAP.MIB

The SNMP Community Strings and SNMP Trap Targets policies utilize SNMP.

To receive SNMP traps on your SNMP management console, you must copy the ZENTRAP.MIB file from the *ZENworks for Servers Program* CD or the ZENworks 6 Server Management Program CD to the location that your management console uses to manage MIBs, then compile it. Your SNMP management console will then be able to receive and interpret SNMP traps from ZfS.

ZENTRAP.MIB is located on the *Program* CD under ZfS\TEDPOL\FILES\MIBS or ZENWORKSFORSERVERS\ZfS\TEDPOL\FILES\MIBS.

Configuring the Container Package Policy

The Search policy is used by the Distributor for information on how to read the eDirectory tree when the Distributor has been refreshed.

IMPORTANT: If you do not use the Search policy, ZfS will search up to [Root] and reread the objects every hour. Be sure to configure and enable the Search policy to limit unnecessary search traffic.

To configure the Search policy:

- 1 In ConsoleOne, right-click the Container Package > click Properties > click the Policies tab.
- 2 Click Search Policy > Properties > Search Level.

If the box under the Enabled column is not checked for the Search policy, click it before clicking Properties. A policy must be enabled to activate the Properties button.

- 3 To determine the upper limits of the search policy, select one of the following:

Search Location	Description
Object Container	Search to the parent container of the Server object
Partition	Search to the Partition Root
Selected Container	Search to the selected container

Search Location	Description
[Root]	Search to the root of the tree

If you chose Selected Container, browse to select the container.

To determine searching limits in either direction of the item selected, enter a number. For example:

#	Description
0	Limits the search to the current level (as set in the Search For Policies Up To field).
1	Limits the search to one level above the current level (as set in the Search For Policies Up To field). For example, if you specify the server's parent container in the Selected Container field, +1 would limit the search to one level above the parent container.
-1	Limits the search to one level below the chosen search level (as set in the Search For Policies Up To field). For example, if you select [Root] in the Search For Policies Up To field, -1 would allow searching up to one level below [Root].

4 To determine the search order, click Search Order.

Type	Description
Object	Server
Group	Server Group
Container	Container of Servers

Use the arrow keys to change the order. You can also click Add or Remove to change which object types are used.

5 For ZfS 2 servers, to set the frequency for refreshing policies from eDirectory, click the Refresh Interval tab > click Policy Manager Will Refresh Policies From eDirectory > select the time increments.

For ZfS 3.0.2 servers, policies are refreshed when they are received at the Subscriber.

You can specify a refresh frequency. The default is once every hour.

If you leave both time increments at zero (days and hours), policies will not be refreshed from eDirectory, even if you have Policy Manager Will Refresh Policies From eDirectory checked.

Changes made to enabled policies are not enforced until they are refreshed at the given refresh interval. However, you can manually refresh all policies using the POLICY REFRESH command at the server console. The refresh rate is listed in seconds at the server console (1 hour = 3600 seconds).

6 Click OK to close the policy.

If you click Cancel, none of the Search policy changes made on any of the tabs will be saved.

- 7** To associate the policy package so that the Search policy will be enforced on the Distributor, click the Associations tab > click Add.
- 8** Browse to select the container where the Distributor object resides (or any container above it) > click OK.

If you click Cancel, the association you made will not be saved.

Configuring Server Package Policies

For ZfS 3.0.2 Policy and Distribution Services, the policies similar to those contained in the Server Package must be distributed for enforcement. Because the policies contained in the Server Package must be associated for enforcement, you must use the policies in the Distributed Server Package for ZfS 3.0.2 Policy and Distribution Services servers. For more information, see “[Configuring Distributed Server Package Policies](#)” on page 475.

For instructions on the policies contained in the Server Package that are applicable to ZfS 2 servers, see the ZfS 2 documentation on the [Novell Documentation Web site \(http://www.novell.com/documentation/lg/zfs2/index.html\)](http://www.novell.com/documentation/lg/zfs2/index.html).

For ZfS 3.0.2, only Server Inventory uses the ZENworks Database policy contained in the Server Package. For more information, see “[Configuring the Database Location Policy](#)” on page 710.

Configuring Service Location Package Policies

Because the Distributor does not receive Distributions, policies for a Distributor must be associated with the container where its object resides. The Service Location Package contains policies used by the Distributor.

The ZENworks for Servers License policy only applies to ZfS 2 servers. For more information, see the ZfS 2 documentation on the [Novell Documentation Web site \(http://www.novell.com/documentation/lg/zfs2/index.html\)](http://www.novell.com/documentation/lg/zfs2/index.html).

To configure Service Location Package policies, review the following sections:

- ◆ “[SMTP Host](#)” on page 469
- ◆ “[SNMP Trap Targets](#)” on page 470
- ◆ “[Tiered Electronic Distribution](#)” on page 471
- ◆ “[ZENworks Database](#)” on page 474

SMTP Host

Sets the TCP/IP address of the SMTP relay host that processes outbound Internet e-mail. This policy must be enabled if you select the E-Mail option for notifying or logging messages for the Distributor.

To configure the SMTP Host policy:

- 1** In ConsoleOne, right-click the Service Location Package > click Properties.
- 2** Click the SMTP Host policy > click Properties.

If the box under the Enabled column is not checked for the SMTP Host policy, click it before clicking Properties. A policy must be enabled to activate the Properties button.

- 3** Enter the TCP/IP address or DNS name of the relay host server > click OK.

- 4** To associate the policy package so that the SMTP Host policy will be enforced on the Distributor, click the Associations tab > click Add.
- 5** Browse to select the container where the Distributor object resides (or any container above it) > click OK.

If you click Cancel, the association you made will not be saved.

SNMP Trap Targets

Use this property page to establish the targets (or locations) where you want SNMP traps sent from the Distributor. Each target must be a valid TCP/IP address or DNS name.

In ZfS 2, both this policy and the SNMP Trap Target Refresh policy are used to manage SNMP. In ZfS 3.0.2, this policy must be scheduled, because a new Schedule tab on the policy replaces the ZfS 2 SNMP Trap Target Refresh policy.

To configure the SNMP Trap Targets policy:

- 1** In ConsoleOne, right-click the Service Location Package > click Properties.
- 2** Click the SNMP Trap Targets policy > click Properties.
If the box under the Enabled column is not checked for the SNMP Trap Targets policy, click it before clicking Properties. A policy must be enabled to activate the Properties button.
- 3** To add items to the SNMP Trap Targets list on the SNMP Trap Policy tab, click Add.
- 4** On the SNMP Target dialog box, enter valid a TCP/IP address or DNS name > click OK.
- 5** Repeat **Step 3** and **Step 4** for each trap target to be added.
- 6** To schedule the policy, click the Schedule tab > select a type in the Schedule Type field > configure the schedule:
 - “Daily” on page 571
 - “Event” on page 571
 - “Interval” on page 571
 - “Never” on page 572
 - “Package Schedule” on page 572
 - “Relative” on page 572
 - “Run Immediately” on page 572
 - “Time” on page 573
 - “Weekly” on page 573
 - “Monthly” on page 571
 - “Yearly” on page 573
- 7** Click OK when finished.
- 8** To associate the policy package so that the SNMP Trap Targets policy will be enforced on the Distributor, click the Associations tab > click Add.
- 9** Browse to select the container where the Distributor object resides (or any container above it) > click OK.

If you click Cancel, the association you made will not be saved.

Tiered Electronic Distribution

This policy allows you to set default values for the attributes of Distributors and Subscribers. The default values become effective when you associate the Service Location Package to a container above where the Distributor or Subscriber object resides.

If you made changes to any default values for Distributors or Subscribers during installation of ZFS, the Use Policy check box will not be checked when the policy package is associated with the Distributors' or Subscribers' containers.

If you did not make changes to any default values during installation, the Use Policy check box will be checked when the policy package is associated with the Distributors' or Subscribers' containers, and the values in the Tiered Electronic Distribution policy will be used for the Distributor and Subscriber attributes.

IMPORTANT: The Tiered Electronic Distribution policy replaces, not supplements, the similar fields in a Distributor's or Subscriber's properties. Therefore, if you create a Tiered Electronic Distribution policy, make sure you fill in all of the fields on every tab in the policy that you will want to be applied to the affected Distributors or Subscribers. For example, if your Subscriber has a working directory entered in its object's properties, you don't enter a working directory in the Tiered Electronic Distribution policy, then later apply the policy by clicking the Use Policy check box on the Subscriber's properties, the Subscriber will no longer have a working directory available to it.

To configure the Tiered Electronic Distribution policy:

- 1** In ConsoleOne, right-click the Service Location Package > click Properties.
- 2** Click the Tiered Electronic Distribution policy > click Properties.

If the box under the Enabled column is not checked for the Tiered Electronic Distribution policy, click it before clicking Properties. A policy must be enabled to activate the Properties button.

- 3** On the General Settings tab, fill in the fields:

Input Rate: Sets the default input rate to minimize network traffic for TED objects. This determines the receive rate for Subscribers and Distributors. The default value is the maximum that the connection can handle. You can use this rate to control the use of narrow bandwidth links.

Output Rate: Sets the default output rate to minimize network traffic for TED objects. This determines the send rate for Distributors and Subscribers. The default value is the maximum that the connection can handle. There are three output priorities where you can specify a rate:

- ◆ **High Priority:** These Distributions will be sent before any Medium or Low priority Distributions.
- ◆ **Medium Priority:** These Distributions will be sent after all High priority and before any Low priority Distributions.
- ◆ **Low Priority:** These Distributions will be sent after all High and Medium priority Distributions.

For more information, see [“Prioritizing Distributions” on page 410](#).

Maximum Concurrent Connections: Specifies a default maximum number of Distribution threads that can be running concurrently for Distributors and parent Subscribers. The default value is unlimited (blank field).

This number can help in load-balancing on a Distributor's sending activity and spread network traffic over an entire scheduling window.

Connection Time-out: Specifies a default number of seconds before the Distributor times out when connecting to another node, or specifies the number of seconds a Subscriber will wait for a response from a Distributor (receiving) or a Subscriber (sending) before ending the connection.

After the time has transpired, a Distributor will end the connection and not retry until the Channel's Send schedule starts again. If a connection is ended during sending or receiving, a Subscriber will not start again until the next time the Channel's Send schedule starts.

The default value is 300 seconds (five minutes). The available range in seconds is 1 to 60,000. You should select a reasonable time to wait for a response from one node to another.

IMPORTANT: This interval must be increased on slow or busy links where longer delays are frequent.

Working Directory: Enter a default TED directory to store Distributions, persistent status, and temporary files on a server. The directory needs to be located where there is enough free space to handle processing of Distributions.

The Working Directory field allows the use of variables to specify the volume/drive and directory names. However, variables will only work with Subscribers.

IMPORTANT: Distributors are not able to resolve variables and will use exactly what is specified in the Working Directory field. For example, if the value was %VOL%TED1\WORKING, the Distributor would create a working directory on the SYS: volume named SYS:\%VOL%\TED\WORKING, because it could not resolve %VOL%.

For more information, see [“Working Directories” on page 447](#).

Parent Subscriber: Subscribers should generally not receive their Distributions directly from a Distributor. You can browse for a Subscriber to be the default parent Subscriber for your whole network that will pass on Distributions when a Subscriber object might not have a parent Subscriber defined in its properties.

Disk Space Desired To Be Left Free: Use this as the default value to ensure there will be enough free disk space for receiving Distributions where you might not have this value defined in a Subscriber object's properties. A Subscriber will not attempt to receive a Distribution if the disk space value set here is insufficient.

4 Click the General tab > click Messaging > fill in the fields:

Server Console: Procedure to follow when displaying messages at the server console. The default is Level 4 (Information & Level 3 Messages).

SNMP Trap: Procedure to follow when sending SNMP traps. The default is Level 0 (No Messages).

Log File: Procedure to follow when recording information to a log file. The default is Level 5 (Trace Information & Level 4 Messages).

Filename: By default, this field is blank. Whatever log filename you select, it replaces TED.LOG for the servers where this policy is enforced.

To create a log file, enter the log file's filename using the following format:

installation_path\directory_pathfilename.filename_extension

The *installation_path* is not required for ZENworks to locate the log file, but it is easier for you to locate the file if the path is included.

IMPORTANT: Because the log file can become quite large, for NetWare servers we recommend that you do not use the SYS: volume.

Some filename examples:

DIST.LOG

TEDLOG.TXT

Delete Log Entries Older Than __ Days: Controls disk space usage. For log files, it is important to set the message levels at minimal detail and to purge entries older than six days (the default).

E-Mail: Procedure to follow when sending e-mail messages. None or Errors Only are recommended to minimize unnecessary e-mail traffic. The default is Level 0 (No Messages).

Users: Add users, groups, or e-mail addresses.

Address Attribute: Displays the attribute of the associated user or group. You can change the attribute from the drop down list, which displays over three dozen options.

Following are some of these options:

CN	Mailbox ID
Description	NSCP:mailHost
EMail Address	OU
Full Name	Physical Delivery Office Name
Employee ID	Postal Code
Entrust:User	Postal Office Box
Generational Qualifier	Surname
Given Name	Telephone Number
Initials	Title
Internet EMail Address	uniqueID

- 5** To assign default values to variables used by the Subscriber, click the Variables tab > click Add > fill in the fields:

Variable: Name of the variable. It should indicate how the variable will be used. For example, WORKINGVOL.

The variable name can be derived from predefined and user-defined variables.

Value: The value that the Subscriber will use when this variable is specified. For example, DATA:.

A value can be another variable name. You can nest variables using this method.

To ensure that extraction will take place, provide an absolute path to the Subscriber. For example, if the path is only the DATA volume, make sure the colon (:) is included, because it is a necessary part of the full path.

Description: Describes how the variable will be used. For example:

Volume for the working directory.

Note that if a variable defined here does not exist in a Subscriber's variables list, it will automatically be added. However, if the variable does exist in the Subscriber's variables list, the definition in the Subscriber will prevail.

- 6** To assign a default refresh schedule for all Distributors, click the Schedule tab > click Distributor Refresh Schedule > select a schedule in the Schedule Type field > configure the schedule:

- “Never” on page 572
- “Daily” on page 571
- “Monthly” on page 571
- “Yearly” on page 573
- “Interval” on page 571
- “Time” on page 573

For information on the refresh schedule, see [“TED Object Schedules” on page 574](#).

IMPORTANT: We recommend the Distributor’s Refresh schedule be daily, unless changes to Distributions warrant a more frequent refresh. However, do not refresh the Distributor more often than every five minutes. The following can need up to five minutes to complete their processes: Distribution building, eDirectory replication, and tree walking (when no Search policy is defined).

- 7** To assign a default extraction schedule for all Subscribers, click the Schedule tab > click Subscriber Extract Schedule > select a schedule in the Schedule Type field > configure the schedule:

- “Never” on page 572
- “Daily” on page 571
- “Monthly” on page 571
- “Yearly” on page 573
- “Interval” on page 571
- “Time” on page 573
- “Run Immediately” on page 572

For information on the extraction schedule, see [“TED Object Schedules” on page 574](#).

- 8** Click OK to close the policy.
- 9** To associate the policy package so that the Tiered Electronic Distribution policy will be enforced on the Distributor, click the Associations tab > click Add.
- 10** Browse to select the container where the Distributor object resides (or any container above it) > click OK.

If you click Cancel, the association you made will not be saved.

ZENworks Database

Sets the DN for locating a ZENworks Database object. This policy must be in effect for ZfS to locate a database file for logging successes and failures that are used in creating reports. If a database object is not identified with this location policy, ZfS will not use the corresponding database file to log reporting information. Therefore, you should create this policy for each database object in the tree.

Use this property page to select the database object that will be associated with the current ZENworks Database policy. The policy will not be in effect until you have distributed the policy to the Subscribers, or associated the policy with the Distributor.

The ZENworks database is used to store reporting information for Distributions and Server Policies.

To configure the ZENworks Database policy:

- 1** In ConsoleOne, right-click the Service Location Package > click Properties.
- 2** Click the ZENworks Database policy > click Properties.

If the box under the Enabled column is not checked for the ZENworks Database policy, click it before clicking Properties. A policy must be enabled to activate the Properties button.

- 3** Click the Policy/Distribution Management tab.
- 4** In the Database DN field, browse for the ZENworks Database object that represents the database for this policy > click OK.
- 5** To associate the policy package so that the ZENworks Database policy will be enforced on the Distributor, click the Associations tab > click Add.
- 6** Browse to select the container where the Distributor object resides (or any container above it) > click OK.

If you click Cancel, the association you made will not be saved.

Configuring Distributed Server Package Policies

You can configure Distributed Server Package policies to automate control of various server behaviors and processes and to automate control of SMTP Host TCP/IP addresses, SNMP Trap Targets, and the ZENworks database DN.

There are several Policies tab options for server policies, one for each supported operating system. The policies that are available on the General tab apply to servers on all platforms. The policies available on the specific platform tabs apply only to the servers for those platforms.

Platform-specific policies, such as those on the NetWare tab, always override similar policies on the General tab for a particular policy package.

All policies are contained in the NetWare policies. Therefore, only the NetWare policies are documented here. The information applies equally to each platform.

To configure Distributed Server Package policies, review the following sections:

- ◆ [“Copy Files” on page 475](#)
- ◆ [“NetWare SET Parameters” on page 477](#)
- ◆ [“Scheduled Down” on page 478](#)
- ◆ [“Scheduled Load/Unload” on page 478](#)
- ◆ [“Server Down Process” on page 479](#)
- ◆ [“Server Scripts” on page 481](#)
- ◆ [“SMTP Host” on page 482](#)
- ◆ [“SNMP Community Strings” on page 482](#)
- ◆ [“SNMP Trap Targets” on page 482](#)
- ◆ [“Text File Changes” on page 483](#)
- ◆ [“ZENworks Database” on page 484](#)
- ◆ [“ZENworks for Servers” on page 485](#)

Copy Files

The Copy Files policy enables copying of files on a server from one location to another by using policy configurations. You can either copy or move the files.

To configure the Copy Files policy:

- 1** In ConsoleOne, click the Distributed Server Package's container > right-click the Distributed Server Package > click Properties.
- 2** Click the Policies tab > select the platform from:
 - General
 - Windows
 - NetWare
 - Linux
 - Solaris

3 Click Add > click Copy Files > enter a policy name > click OK.

4 Click Properties.

The Copy Files tab displays.

5 Click Add.

Local File Copy #1 defaults. You can edit that name.

6 Fill in the fields:

Source Path: Enter the full path where the files to be copied are located.

You can use wildcards in the path:

* = any number of characters

? = any single character in that position

??? = any characters in those positions

Target Path: Enter the full path where the copied files are to be placed.

You can use wildcards in this path. This path does not need to mirror the source path.

However, you could mirror an existing target path.

Include Subdirectories: Includes all subdirectories and their files beginning from the directory at the end of the path; otherwise, only the files in the directory at the end of the path will be copied.

Maintain Attributes: Maintains the file attributes in the target's file system that exist in the source's file system.

Overwrite Destination Files: Overwrites files of the same name in the destination directories, regardless of differences in file dates. If you do not enable this option, files of the same name will not be replaced.

Maintain Trustees: Maintains the file's trustee attributes.

When a File Is Locked: Select one or both:

- ◆ **Retry __ Times:** Retries overwriting a locked file the number of times you select before failing to replace the file. Leave this check box unchecked to not replace locked files on the target file system.
- ◆ **Kill Connection of Open Files:** Attempts to kill the connection of locked files so they can be overwritten. This applies only to files being extracted, not to files being accessed to build the Distribution. If a file belonging to a Distribution is locked when the Distribution is being built, the build will fail. Server and NLM connections cannot be killed.

Error Processing: Fail On Error is checked by default. This stops the file copying process when an error is encountered in copying. To continue file copying when an error is encountered, click Continue On Error.

Operation: Sets whether to copy or move the files identified in the Source Path.

- 7** Click the Schedule tab > schedule the policy (see “[Scheduling Policies](#)” on page 488).
- 8** Click OK to close the policy.

NetWare SET Parameters

You can automate the use of SET parameters by your servers.

To configure NetWare SET parameters:

- 1** In ConsoleOne, click the Distributed Server Package's container > right-click the Distributed Server Package > click Properties.
- 2** Click the Policies tab > NetWare (or General).
- 3** Click Add > select NetWare Set Parameters.
- 4** Enter a name for this SET parameters policy > click OK.

Because the policies selected from this dialog box are plural, you can have multiple SET parameter policies listed on the Policies tab. Therefore, enter a unique name for this policy.

When you click OK after naming the SET parameters policy, it will be checked and selected on the Policies tab.

- 5** Click Properties.

The Set Commands tab displays.

- 6** Click Add.

The NetWare Server SET Command Wizard opens.

- 7** Select the server containing the SET parameters > click Next.

IMPORTANT: The Policy/Package Agent must be running.

- 8** Select all of the commands you want to configure in the policy.

You can select whole categories by clicking the check box for the category, or clicking the plus sign to expand a SET command category and clicking the check boxes for individual commands to be included.

WARNING: Do not select the Set Developer Option SET command and change the default of Off to On. This parameter is meant to help developers debug server abends. It disables some of the operating system checking to prevent certain abends from occurring. Also, if the Set Developer Option is turned On, running NCP™ scripts that require keyboard entry could abend the server.

- 9** Click Finish when you are finished selecting the commands.

The selected commands are now displayed in the Set Commands tab for the policy.

- 10** To edit a SET command, click its plus sign to expand its attributes.

- 11** To edit an attribute, click the attribute > click Edit.

A dialog box is displayed in which you can make changes to the attribute.

- 12** Repeat the previous step for each attribute to edit for a given SET command.

- 13** Repeat [Step 10](#) through [Step 12](#) to edit another SET command's attributes.

14 Schedule the policy (see “[Scheduling Policies](#)” on page 488).

15 Click OK to close the policy.

If you click Cancel, neither the schedule nor the SET parameter changes will be saved.

Scheduled Down

You can automate when and how you want a server to go down, and whether it should be automatically brought back up.

To configure a scheduled downing for a server:

1 In ConsoleOne, right-click Distributed Server Package > click Properties.

2 Click the Policies tab > NetWare (or other platform).

3 Click Add > select Scheduled Down.

4 Enter a unique name for the policy > click OK.

Because the policies selected from this dialog box are plural, you can have multiple Scheduled Down policies listed on the Policies tab. Therefore, enter a unique name for this policy.

When you click OK after naming the Scheduled Down policy, the policy will be checked and selected on the Policies tab.

5 Click Properties.

The Up Procedure tab displays.

6 Select the downing method:

Downing Option	Description
Reset Server	Downs the server and then does a warm boot
Restart Server	Downs the server and then restarts it
Down Server	Downs the server, does not restart it

7 Schedule the policy (see “[Scheduling Policies](#)” on page 488).

8 Click OK to close the policy.

If you click Cancel, neither schedule for your newly scheduled Down policy will be saved.

Scheduled Load/Unload

You can automate scheduled loading and unloading of NLM files and Java Class processes, and Linux and Solaris executables.

To configure the schedules:

1 In ConsoleOne, click the Distributed Server Package's container > right-click the Distributed Server Package > click Properties.

2 Click the Policies tab > NetWare (or other platform).

3 Click Add > select Scheduled Load/Unload.

4 Enter a name for this Load/Unload policy > click OK.

Because the policies selected from this dialog box are plural, you can have multiple Load/Unload policies listed on the Policies tab. Therefore, enter a unique name for this policy.

When you click OK after naming the Load/Unload policy, it will be checked and selected on the Policies tab.

5 Click Properties.

The Scheduled Load/Unload tab displays.

6 Click Add.

7 Select one of the following options:

“Load NLM/Process” on page 623

“Load Java Class” on page 623

“Unload Process” on page 624

“Start Service” on page 624

“Stop Service” on page 624

Click an item for further instructions on configuring it.

8 Repeat **Step 6** and **Step 7** for each NLM or process to be included.

9 To rearrange the order, use the arrow keys.

10 Schedule the policy (see “Scheduling Policies” on page 488).

11 Click OK to close the policy.

If you click Cancel, your newly scheduled Load/Unload policy will not be saved.

Server Down Process

You can automate the procedures your servers use when they are downed.

To configure the downing process for a server:

1 In ConsoleOne, click the Distributed Server Package's container > right-click the Distributed Server Package > click Properties.

2 Click the Policies tab > NetWare (or other platform).

3 Click the Server Down Process policy > click Properties.

If the box under the Enabled column is not checked for the Server Down Process policy, click it before clicking Properties. A policy must be enabled to activate the Properties button.

4 To configure procedures for downing, click the Down Procedure tab > Down Procedures.

5 To enable the policy's options, check the box labeled Follow This Procedure When a Down Server Is Triggered > enter the number of minutes to wait before downing the server.

6 To disable login before downing, check the box > enter the number of minutes before downing to disable login.

7 To drop connections before downing, check the box > enter the number of minutes before downing the server to drop connections.

8 To configure an order for unloading, click the Down Procedure tab > click Ordered Unload.

8a To include NLM files and processes, check the Unload These NLMs and Kill These Processes in This Order Before Downing box.

8b Click Add.

- 8c** Select either NLM or Process > enter the name > click OK.
- 8d** To change the order, use the arrow keys.
- 9** To configure reporting, click the Notification tab > Reporting.
- 9a** To have another server send an SNMP alert if the server is not up after a specified time, check the Send SNMP Alert box > enter the number of minutes.
- For information about displaying SNMP traps on your management console, see [“Compiling ZENTRAP.MIB” on page 467](#).
- 9b** To specify which servers can watch for the restart and send the alert in case of failure, click Add to display an ordered list of candidate servers.
- Policy and Distribution Services starts at the top of the list to communicate with the first server and use it for the alert notification. If Policy and Distribution Services cannot communicate with a server, the next one on the list is tried. The first server that can be used will be the one that is scheduled to send the alert.
- 9c** Browse to select a server.
- 9d** Repeat [Step 9a](#) through [Step 9c](#) for each server needed.
- 9e** To change the order, use the arrow keys.
- 10** To configure broadcast messages, click the Notification tab > Broadcast Messages > Send Messages To Connected Users.
- 10a** Enter the number of times to send the message.
- 10b** To broadcast custom text, enter it in the box.
- 10c** To include the predefined message containing a time as the last line of your broadcast, check the box.
- The x minutes is derived from dividing the number of times from [Step 10a](#) into the number of minutes left before the server will be downed, then subtracting that amount (in whole minutes) for the amount to display in each broadcast. For example, if there are 10 minutes left and you select 5 in [Step 10a](#), the message will be broadcast every two minutes. The number of minutes left after each broadcast will be two minutes less than at the last broadcast.
- 11** To configure targeted messages, click the Notification tab > Targeted Messages > Send E-mail To Selected Users When Server Is Going Down.
- 11a** To specify the users, groups, or e-mail addresses to receive the targeted messages, click Add.
- 11b** Select either User, Group, or E-Mail Address.
- 11c** Browse to select the user or group, or enter the e-mail address.
- 11d** Repeat [Step 11a](#) through [Step 11c](#) for other users, groups, or e-mail addresses.
- 12** To configure the conditions for downing a server, click the Conditions tab > Use Conditions.
- 12a** To enter the conditions, click Add.
- 12b** Select from the following conditions to specify when not to bring the server down:
- Some of these conditions require you to enter valid names. Others use the Select Object dialog box to browse for them.
- File Open:** If the files that you specified are open. For example, an .EXE.

NLM Loaded: If the NLM files that you specified are running.

Server Connected: If the server that you specified is connected.

User Connected: If the users that you specified are connected.

Number of User Connections: If the number of users connected exceeds the number you specify. In other words, don't bring the server down if too many users would be affected.

Workstation Connected: If the workstations that you specified are connected.

12c Repeat **Step 12a** and **Step 12b** for each condition to add to the list.

12d To change the order, use the arrow keys.

13 Click OK to close the policy.

If you click Cancel, none of the Server Down Process policy changes made on any of the tabs will be saved.

Server Scripts

You can automate script usage by your NetWare servers.

To configure server scripts:

1 In ConsoleOne, click the Distributed Server Package's container > right-click the Distributed Server Package > click Properties.

2 Click the Policies tab > NetWare (or other platform).

3 Click Add > select Server Scripts.

4 Enter a unique name for the policy.

Because the policies selected from this dialog box are plural, you can have multiple Script policies listed on the Policies tab. Therefore, enter a unique name for this policy.

When you click OK after naming the Script policy, it will be checked and selected on the Policies tab.

5 Click Properties.

The Script tab displays.

6 Click Add > select Server Scripts.

7 Enter a script name.

Script #1 displays.

8 Select the script type (NCF, NetBasic*, PERL).

9 Enter the script text.

10 Repeat **Step 6** through **Step 9** for each script to be added.

11 Use the arrow keys to arrange the order to execute the scripts.

12 Schedule the policy (see "**Scheduling Policies**" on page 488).

13 Click OK to close the policy.

If you click Cancel, neither the schedule nor any of the scripts entered will be saved.

SMTP Host

You can set the TCP/IP address of the relay host that processes outbound Internet e-mail.

To configure the SMTP Host policy:

- 1 In ConsoleOne, right-click the Service Location Package > click Properties.
- 2 Click the SMTP Host policy > click Properties.

If the box under the Enabled column is not checked for the SMTP Host policy, click it before clicking Properties. A policy must be enabled to activate the Properties button.

The SMTP Host tab defaults.

- 3 Enter the TCP/IP address or DNS name (such as mail.novell.com) > click OK to close the policy.

If you click Cancel, the TCP/IP address will not be saved.

SNMP Community Strings

This policy provides configuration and scheduling of SNMP community strings.

To configure the SNMP Community Strings policy:

- 1 In ConsoleOne, click the Distributed Server Package's container > right-click the Distributed Server Package > click Properties.
- 2 Click the Policies tab > NetWare (or other platform).
- 3 Click the SNMP Community Strings policy > click Properties.

If the box under the Enabled column is not checked for the SNMP Community Strings policy, click it before clicking Properties. A policy must be enabled to activate the Properties button.

The SNMP Community Policy tab displays.

- 4 Fill in the Community Strings fields:

Monitor
Control
Trap

Community Strings are case sensitive. Enter a string for each field as needed.

- 5 Click the Schedule tab > schedule the policy (see [“Scheduling Policies” on page 488](#)).
- 6 Click OK to close the policy.

SNMP Trap Targets

You can set targets for SNMP traps for the Subscriber Agent and Policy/Package Agent.

For information about displaying SNMP traps on your management console, see [“Compiling ZENTRAP.MIB” on page 467](#).

Understanding How the Windows Trap Target Policy Enforcer Behaves

The following abbreviations are used in this section to represent these Windows registry locations:

- ♦ **AGENT_KEY:** HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\SNMP\Parameters

- ♦ **ZFS_KEY:** HKEY_LOCAL_MACHINE\SOFTWARE\Novell\Zenworks for Servers

The Windows SNMP trap target policy enforcer performs in the following sequence:

1. The policy enforcer first verifies an installation of an SNMP agent. This is done by checking if AGENT_KEY exists. If it exists, the enforcer assumes that an SNMP agent is installed and continues with the following steps. Otherwise, an error is returned and the processing stops.
2. The enforcer keeps track of all trap targets added by the ZfS policy by placing the trap targets in ZFS_KEY. The trap targets are organized like the trap targets in AGENT_KEY with a subkey of TrapConfiguration. The subkey TrapConfiguration contains community strings that are represented as registry subkeys. These community strings contain the trap target values associated with each community string.
3. Each trap target in the ZfS policy is put into AGENT_KEY, unless it already exists. The policy enforcer ensures that each ZFS trap target is found, or is added to each community string. If no community strings exist in AGENT_KEY, a community string named "public" will be created.
4. Any previously added trap targets found in ZFS_KEY that are removed from the ZfS policy are removed from AGENT_KEY. Trap targets not added by ZfS will not be removed.
5. If Microsoft's SNMP agent is installed, the agent's trap targets are automatically updated with registry changes.

Configuring the SNMP Trap Target Policy

To configure the SNMP Trap Targets policy:

- 1** In ConsoleOne, right-click the Service Location Package > click Properties.
- 2** Click the SNMP Trap Targets policy > click Properties.
If the box under the Enabled column is not checked for the SNMP Trap Targets policy, click it before clicking Properties. A policy must be enabled to activate the Properties button.
- 3** Click Add.
- 4** Enter a new target > click OK.
TIP: Enter the TCP/IP address or DNS name of the target server. IPX addresses are not supported.
- 5** Repeat **Step 3** through **Step 4** for each new trap target.
- 6** Click the Schedule tab > schedule the policy (see “**Scheduling Policies**” on page 488).
- 7** Click OK to close the policy.

If you click Cancel, none of the targets that you entered will be saved.

Text File Changes

You can automate changes to text files on your servers.

To configure text file changes:

- 1** In ConsoleOne, click the Distributed Server Package's container > right-click the Distributed Server Package > click Properties.
- 2** Click the Policies tab > NetWare (or other platform).
- 3** Click Add > select Text File Changes.
- 4** Enter a unique name for the policy.

Because the policies selected from this dialog box are plural, you can have multiple text file policies listed on the Policies tab. Therefore, enter a unique name for this policy.

When you click OK after naming the text file policy, it will be checked and selected on the Policies tab.

5 Click Properties.

The Text Files tab defaults.

6 Click Add.

After one text file has been added, you will be given the opportunity to select whether you are adding another text file or another change item for the selected text file.

To add another text file, select Text File. It does not matter which text file or change item is selected in the left pane—the text file will be added to the far left level.

To add another change to a text file, in the left pane click the text file for the change > click Add > select Change. The change item will be added under the selected text file.

7 If you are adding a text file, enter the name of the text file.

8 Accept the default name (such as Change #1) or rename it > if you are adding a text file, click OK.

9 Click the down-arrow for the Change Mode field > select the change mode from the drop-down list.

10 Click the down-arrow for the Search Type field > select the search type from the drop-down list.

11 Enter the exact search string.

12 Check the box if you want the string search to be case sensitive.

13 To find all occurrences of the search string, make sure the box is selected, or deselect the box to find only the first occurrence.

14 Click the down-arrow for the Result Action field > select the action from the drop-down list that should result if a string is matched.

15 If you will be replacing a string or entering a new one, enter the text in the New String text box.

16 Repeat **Step 6** through **Step 15** for each text file to add or each change to be made.

17 To reorder the text files and change items, use the arrow keys.

18 Schedule the policy (see “**Scheduling Policies**” on page 488).

19 Click OK to close the policy.

If you click Cancel, neither the schedule nor any of the text files entered will be saved.

ZENworks Database

If you have installed the ZENworks database, you can set its DN so that the server this policy is associated with can find the database for logging information.

To configure the ZENworks Database policy:

1 In ConsoleOne, right-click the Service Location Package > click Properties.

2 Click the ZENworks Database policy > click Properties.

If the box under the Enabled column is not checked for the ZENworks Database policy, click it before clicking Properties. A policy must be enabled to activate the Properties button.

- 3** Click the Policy/Distribution Management tab.

The Inventory Management tab defaults. Make sure you are using the correct tab.

- 4** Enter the DN of your ZENworks Database object, or browse to select the DN > click OK to close the policy.

If you click Cancel, the DN will not be saved.

ZENworks for Servers

This policy provides basic configuration parameters for Policy and Distribution Services.

To configure the ZENworks for Servers policy:

- 1** In ConsoleOne, click the Distributed Server Package's container > right-click the Distributed Server Package > click Properties.
- 2** Click the Policies tab > NetWare (or other platform).
- 3** Click the ZENworks for Servers policy > click Properties.

If the box under the Enabled column is not checked for the ZENworks for Servers policy, click it before clicking Properties. A policy must be enabled to activate the Properties button.

The General – Status tab displays.

- 4** To determine the policy's general status:
 - 4a** Select the procedure to follow when displaying messages at the server console.
 - 4b** Select the procedure to follow when sending SNMP traps.

For information about displaying SNMP traps on your management console, see [“Compiling ZENTRAP.MIB” on page 467](#).

- 4c** Select the procedure to follow when recording information to a log file.

Logging Procedure	Description
Log File	<p>Check to enable and enter the log file's filename. Include its full path. By default, Policy and Distribution Services uses ZENWORKS\PDS\SMANAGER\DEFAULTLOG.TXT, unless you enter a filename here. Then, for the servers where this policy is enforced, the log file you specify here is used instead of DEFAULTLOG.TXT.</p> <p>Until Policy and Distribution Services has loaded and read eDirectory, it temporarily uses ZFSINIT.LOG for logging.</p> <p>Some examples:</p> <pre>sys:\zenworks\smanager\polpack.txt sys:\zenworks\smanager\polpack.log data:\zenworks\policies.log</pre>
Delete Log Entries Older Than __Days	Use this option to control disk space usage.
E-Mail Messages	Select whether to send e-mail messages. The None or Errors Only options are recommended.

Logging Procedure	Description
◆ Users	You can add users, groups, or e-mail addresses.
◆ Address Attribute	After you select users or groups, this field displays the attribute of the associated user or group. You can change the attribute from the drop-down list.

IMPORTANT: Set the E-Mail Messages option to either None or Errors Only. If you set this to a more detailed level, performance will degrade because of the extra e-mail messages that will be created.

5 To determine the policy’s configuration, click the ZENworks for Servers tab > Configuration.

5a Enter a console prompt.

You can customize the prompt using plain text and variables. The default is:

```
%SERVER_DN% - ZFS>
```

You can use any of the predefined or user-defined variables (for more information, see “Types of Variables” on page 580).

5b Enter a working path.

This is for Policy and Distribution Services temporary and backup files. The default directory is ZENWORKS\PDS\SMANAGER\WORKING.

5c To determine how old database information should be before purging, enter the number of days.

All Policy and Distribution Services information older than the number of days entered will be purged when ZfS is started on the same server where ZFSLOG.DB resides.

IMPORTANT: The database can only be purged if ZfS is running on the same server where ZFSLOG.DB is located.

6 Click OK to close the policy.

If you click Cancel, none of the policy changes on any of the tabs will be saved.

Creating Custom Log Files Using Policies

If you want to create custom log files, you can use either the Tiered Electronic Distribution policy (Service Location Package) or the ZENworks Server Management policy (Distributed Server Package):

- ◆ **Tiered Electronic Distribution policy:** With this policy, you associate its Service Location Package to an eDirectory container, and all Distributor and Subscriber objects under it can use this policy. The Use Policy check box that is displayed in each of the object’s properties allows you to individually select whether that Distributor or Subscriber should use the policy. The check box is disabled by default.

Using this policy, the Distributor Agent and Subscriber Agent log TED information to your custom log file for the selected Distributors and Subscribers.

- ◆ **ZENworks Server Management policy:** With this policy, you distribute its Distributed Server Package to the servers where you want the policy enforced.

Using this policy, the Policy/Package Agents for these servers log policy and software package information to your custom log file.

When creating and configuring one of these policies, the Path and Filename field for the log file is blank by default.

For information on how to create and configure these policies, see:

- ◆ [“Tiered Electronic Distribution” on page 471](#)
- ◆ [“ZENworks for Servers” on page 485](#)

Enabling Policies

A policy must be enabled before it can be in effect for the policy package. You can disable a policy without removing it from the package.

To enable a policy:

- 1** In ConsoleOne, right-click the Policy Package object containing the policy to be enabled > click Properties.
- 2** To enable a policy, click its check box under the Enabled column.
If you enable a policy, make sure it is correctly configured.
- 3** To cause an enabled policy to be enforced, distribute the policy package.
For more information, see [“Distributing Policies” on page 487](#).

Distributing Policies

You must distribute a distributed policy package before its policies can be in effect. When you do distribute the package, its enabled policies will only be in effect for the server where it is distributed after the Subscriber has extracted the Distribution.

To distribute policies to a server:

1. Create a TED Distribution that is a Policy Package type.
2. Configure the policies in the policy package.
3. Select a Channel for the Policy Package Distribution.
4. Subscribe the Subscribers to the selected Channel.
5. Send the Distribution.

The Policy/Package Agent on the receiving server will extract the enabled policies and enforce them on the server.

Associating Policies

Because Distributors do not receive policies through Distributions, the Distributor object needs to be associated with the Container Package object so that it can use the Search policy for how to read the eDirectory tree when the Distributor is refreshed.

The Distributor object also needs to be associated with the Service Location Package. This package contains the ZENworks Database policy so that the Distributor Agent can locate the database file for writing report information. It also contains other policies the Distributor uses (see [“Configuring Service Location Package Policies” on page 469](#)).

For associating policy packages with ZfS 2 objects, the steps are very similar to the following procedures.

To associate policy packages with the Distributor object's container:

- ◆ [“Associating a Policy Package to the Distributor Object” on page 488](#)
- ◆ [“Associating the Distributor Object to a Policy Package” on page 488](#)

Associating a Policy Package to the Distributor Object

To associate a policy package to the Distributor object's container:

- 1** In ConsoleOne, right-click the policy package > click Properties.
- 2** Click the Associations tab > click Add.
- 3** Browse to select the container where the Distributor object resides (or any container above it) > click OK.

If you click Cancel, the association you made will not be saved.

Associating the Distributor Object to a Policy Package

To associate the Distributor object's container with a policy package:

- 1** In ConsoleOne, right-click the container where the Distributor object resides (or any container above it) > click Properties.
- 2** Click the ZENworks tab > Associated Policy Packages > Add.
- 3** Browse to select the policy package > click OK.
If you click Cancel, the association you made will not be saved.
- 4** Repeat [Step 2](#) and [Step 3](#) for additional policy packages to be associated with the Distributor object's container.

Scheduling Policies

All policies will use the default schedule (Package Schedule) unless you change the schedule for a policy. You can also edit the default package schedule.

To schedule a policy or to edit the default schedule, review the instructions in the following sections:

- ◆ [“Scheduling a Policy” on page 488](#)
- ◆ [“Editing the Default Schedule” on page 489](#)

Scheduling a Policy

To schedule an individual policy:

- 1** In ConsoleOne, right-click a Policy Package object > click Properties > click the Policies tab.
- 2** Select a policy > click Properties > click the Policy Schedule tab.
- 3** Select a schedule in the Schedule Type field > configure the schedule:
[“Daily” on page 571](#)

- “Event” on page 571
- “Interval” on page 571
- “Never” on page 572
- “Package Schedule” on page 572
- “Relative” on page 572
- “Run Immediately” on page 572
- “Time” on page 573
- “Weekly” on page 573
- “Monthly” on page 571
- “Yearly” on page 573

IMPORTANT: The Relative and Run Immediately schedules are not available for the Scheduled Down policy.

Editing the Default Schedule

To edit the default package schedule:

- 1** In ConsoleOne, right-click a Policy Package object > click Properties.
- 2** Click Edit.
- 3** Select a schedule in the Schedule Type field > configure the schedule:

- “Daily” on page 571
- “Event” on page 571
- “Interval” on page 571
- “Package Schedule” on page 572
- “Relative” on page 572
- “Run Immediately” on page 572
- “Time” on page 573
- “Weekly” on page 573
- “Monthly” on page 571
- “Yearly” on page 573

Viewing Effective Policies

The procedures for viewing which policies are in effect are different for ZfS 2 and ZfS 3.0.2:

- ♦ “Viewing Effective Policies for ZfS 3.0.2 Servers” on page 489
- ♦ “Viewing Effective Policies for ZfS 2 Servers” on page 490

Viewing Effective Policies for ZfS 3.0.2 Servers

To view which ZfS 3.0.2 policies are in effect for the current server object:

- 1** At the ZfS prompt on the server, type Policy List.
Displays the policies that are currently in effect for the server.

Viewing Effective Policies for ZfS 2 Servers

This section is provided for backwards compatibility information for administrators performing an incremental upgrade from ZfS 2.

To view which ZfS 2 policies are in effect for the current server object:

- 1** In ConsoleOne, right-click the Server object > click Properties.
- 2** Click the ZENworks tab > Effective Policies.
- 3** Click the Effective Policies button.
Displays the policies that are currently in effect for the server.
- 4** To view the properties of a given policy, click the policy > click the Package Properties button.

Changing Policy Enforcement

You might need to change or stop policy enforcement for a particular server or a group of servers.

You can change policy enforcement in several ways:

- ◆ [“Modifying a Policy That Is Being Enforced” on page 490](#)
- ◆ [“Stopping a Specific Policy From Being Enforced” on page 490](#)
- ◆ [“Removing Policy Enforcement for a Specific Subscriber” on page 491](#)
- ◆ [“Stopping Enforcement of a Policy Package Distribution” on page 491](#)

Modifying a Policy That Is Being Enforced

To change a policy that is being enforced:

- 1** In ConsoleOne, right-click the Distributed policy package object containing the policy to be modified > click Properties.
- 2** Modify the policy as needed > click OK to exit the policy package properties.

The next time the Distribution containing this policy package is built, the following transpires:

1. A new version of the Distribution is created because it had changed.
2. The Policy Package Distribution is sent according to the Send schedule of the Channel.
3. The Subscribers subscribed to the Channel will all receive and extract the Policy Package Distribution according to their extraction schedules.
4. The modified policy will be enforced on the Subscribers where the Policy Package Distribution was extracted.

Stopping a Specific Policy From Being Enforced

To stop a specific policy from being enforced:

- 1** In ConsoleOne, right-click the Distributed policy package object containing the policy to be stopped > click Properties.
- 2** Click the policy to be stopped > do one of the following:
 - 2a** Click the check box under the Enabled column to disable the policy.

2b Click Remove to remove the plural policy.

Plural policies can be deleted from the policy package because they were previously added using the Add button.

3 Click OK to save the change and exit the policy package properties.

The next time the Distribution containing this policy package is built, the following transpires:

1. A new version of the Distribution is created because it had changed.
2. The Policy Package Distribution is sent according to the Send schedule of the Channel.
3. The Subscribers subscribed to the Channel will all receive and extract the Policy Package Distribution according to their extraction schedules.
4. The disabled/removed policy will no longer be enforced on the Subscribers where the Policy Package Distribution was extracted.

Removing Policy Enforcement for a Specific Subscriber

If you want to stop a distributed policy from being enforced on a specific Subscriber server, rather than on all Subscribers receiving that Distribution, do the following:

- 1** In ConsoleOne, right-click the Subscriber object > click Properties.
- 2** Click the Channels tab > click the Channel containing the policy to be removed from enforcement > click Remove > click OK.
- 3** Click OK to close the Subscriber object's properties.
- 4** On the Subscriber server's file system, delete the following files:
 - ♦ The Distribution directory containing the policy's Distribution file
 - ♦ The related Policy file (.POL) from the SMANAGER\POLICY directory (which was created when the Policy Package Distribution was extracted)
- 5** Reset the Subscriber server to refresh its policy configuration.

The Subscriber will no longer receive the Policy Package Distribution containing that policy, nor will it continue to enforce the policy previously distributed to the Subscriber.

Stopping Enforcement of a Policy Package Distribution

If you need to stop enforcement of a Policy Package Distribution for all of the Subscribers where it was distributed, you must follow certain steps. Because the policy package was distributed, each Subscriber that received the Distribution will still be able to enforce that policy if you only delete the policy package object.

To stop enforcement, do the following:

- 1** In ConsoleOne, delete the Distribution object for the Policy Package type.

IMPORTANT: If the policy package has other policies that you do not want to stop, then do not delete the package. Instead, just disable the policy that you want to stop.
- 2** On the Subscriber server's file system, delete the .POL file that was created by the Policy Package Distribution.

The .POL file is located under the ZENWORKS\PDS\SMANAGER\POLICIES directory.
- 3** Refresh the policies on each Subscriber.

You can do this from each Subscriber server's console using the Policy Refresh command, or from iManager using the Refresh option.

The policies in the Policy Package Distribution will no longer be enforced on the Subscriber after its policies have been refreshed. The refresh process clears its memory of all policies, then reloads them from the Policy Package Distributions existing in its file system.

18 Server Software Packages

Novell® ZENworks® for Servers (ZfS) provides the Server Software Packages component for managing files and applications on your network. Using software packages, you can automate the installation and upgrading of software on your servers.

The real value in using software packages is to set up processes to be done on a server before and after installation of the package.

The following sections will give you an understanding of how you can benefit from using the Server Software Packages component:

- ◆ [“Software Management through Server Software Packages” on page 493](#)
- ◆ [“Understanding Server Software Packages” on page 493](#)
- ◆ [“Planning Server Software Packages” on page 499](#)
- ◆ [“Setting Up Server Software Packages” on page 505](#)
- ◆ [“Converting Older Server Software Packages to ZfS 3.0.2” on page 521](#)

Software Management through Server Software Packages

Software management is done by creating software packages and distributing them using TED. Software packages can be configured so that a server must meet certain minimum requirements before a package is installed on it. Software packages can consist of multiple software package components.

Each software package component can also be configured so that minimum requirements must be met before that component can be installed on the server.

This planning documentation does not cover software packages because their installation does not require Policy and Distribution Services installation or basic configuration decisions. For instructions on creating and using software packages, see [Chapter 18, “Server Software Packages,” on page 493](#).

For information on converting older .SPK and .CPK files to ZfS 3.0.2, see [“Converting Older Server Software Packages to ZfS 3.0.2” on page 521](#).

Understanding Server Software Packages

Policy and Distribution Services provides the means to automate and standardize the distribution and installation of server files and applications. This includes your ability to standardize NLM™ versions, configuration files, databases, and more. Review the following sections:

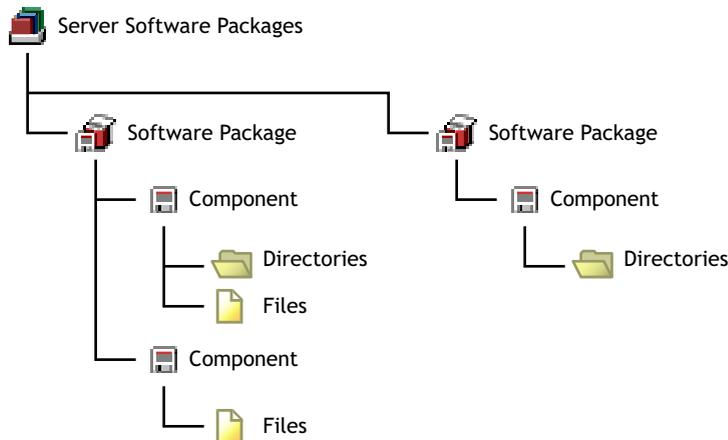
- ◆ [“Understanding Server Software Packages and Components” on page 494](#)
- ◆ [“Understanding Software Package and Component Configurations” on page 494](#)

- ◆ “Determining the Installation Order of Software Packages” on page 495
- ◆ “Compiling Software Packages” on page 496
- ◆ “Accessing Software Packages” on page 496
- ◆ “Distributing Software Packages” on page 497
- ◆ “Distributing Software Packages to a Cluster” on page 498
- ◆ “Rolling Back Software Package Installations” on page 498

Understanding Server Software Packages and Components

To distribute server files and applications for installation on a server, you must include the software in a software package. You create the software packages under the Server Software Packages namespace in ConsoleOne®. Creating software packages is like building a software installation executable.

The following illustrates the relationship between software packages and package components:



Note the following:

- ◆ Software Package objects are displayed under the Server Software Packages namespace
- ◆ A Software Package object can contain multiple Component objects
- ◆ Component objects can contain files and directories
- ◆ Each software package can include all of the files for one or several applications
- ◆ Software Package configuration files (.SPK and .CPK) are stored on a server or workstation file system

Understanding Software Package and Component Configurations

Software packages and their components contain configuration information and installation requirements. Because each Component object is governed by its own set of configuration parameters and installation requirements, you might have multiple components for a software package, such as pre-installation actions, installation actions, and post-installation actions.

You can configure every aspect of the distribution and installation of server files and applications, including the following:

- ◆ Requiring a specific operating system
- ◆ Specifying how much RAM the target server needs
- ◆ Specifying how much disk space the target server needs
- ◆ Requiring certain SET commands on the target server
- ◆ Making changes to the target server's registry
- ◆ Replacing files on the target server
- ◆ Requiring specific PRODUCTS.DAT entries

Software Package Installation Prerequisites

Not only can a software package have installation prerequisites, but each of its components can also have its own installation prerequisites. The hierarchy for adhering to prerequisites to determine installation eligibility is:

- ◆ If the prerequisites for the package are not met, none of the components are installed.
- ◆ If the prerequisites for the package are met, the components are eligible to be installed.
- ◆ If the prerequisites for a component are not met, that component is not installed.

Because some components can be installed while others are not, a partial installation of the software package is possible.

IMPORTANT: When you specify prerequisites, be sure to create prerequisites at the software package level that would apply equally to all of its components, and create prerequisites at the component level that are specific to that component.

Naming Software Packages

When you create a software package, you initially give it a .spk extension, which represents a software package that has not yet been compiled. This file contains all of the installation requirements for the software package and all of its components.

WARNING: Do not use double-byte characters in the software package name. This will cause an error in any report you run on the software package.

Determining the Installation Order of Software Packages

There are two issues concerning the ordering of Server Software Packages in Distributions:

- ◆ [“Forcing the Software Package Distribution Order” on page 495](#)
- ◆ [“How Rollback Is Affected by Software Package Ordering” on page 496](#)

Forcing the Software Package Distribution Order

If you want to include multiple software packages in one Distribution, consider the following:

- ◆ Multiple software packages are not gathered into a Distribution in any particular order when the Distribution is built
- ◆ Multiple software packages are not applied to a server in any guaranteed order when the Distribution is extracted and installed
- ◆ Multiple software packages that are contained in one Distribution and start their installations in a certain order might not all finish in that same order

To install software packages in a particular order:

- 1** Place each software package in its own Distribution (one software package per Distribution).
- 2** Control the order of software package installations by scheduling the order when the Distributions are sent and extracted.

How Rollback Is Affected by Software Package Ordering

Rollback is also affected by the fact that multiple software packages contained in one Distribution won't necessarily finish extracting in the same order that they started.

Although you can specify the order for processing software packages that are contained in a Distribution, this order is not guaranteed. This is because the length of time it takes for software packages to finish processing can be different for each package, and it is the finishing time for a software package that determines its rollback order.

In other words, you can only roll back the last software package that was successfully processed, and then other software packages only in the reverse order of when they finished processing.

You can use the Package List command to view the order in which software packages finished processing. An asterisk marks the next package that is available for rollback.

Compiling Software Packages

After you have defined your software packages, including configuring the components, you must compile the software package. This process compresses the files and applications and their configurations into one file for distribution.

The default extension for a compiled software package is .CPK. The compiled version contains all of the files necessary to install the files and applications that the software package represents.

IMPORTANT: If you enter the path and filename of the .SPK when you are prompted for the compiled filename, the .SPK will be overwritten and can no longer be edited. Be sure to use the .CPK extension when naming the compiled version.

.CPK files have the potential to be very large (hundreds of megabytes), because software packages can include many large files to be copied. Therefore, .CPK files should generally be stored on a server where you have sufficient free disk space.

However, software packages can perform simple functions, which would make the .CPK files' sizes relatively small, so that you could store them on a workstation. For example, a software package could be configured to just delete directories on a file server (see [Appendix F, "Using Server Software Packages to Delete Directories on Servers,"](#) on page 633).

When a rollback-enabled software package is successfully installed, a rollback package is created on the server. Processing this rollback package will return the server to its original state (before the package was installed). For more information, see ["Rolling Back Software Package Installations"](#) on page 498.

Accessing Software Packages

Because the Server Software Packages component uses a namespace in ConsoleOne, it enables you to have access to software packages from any workstation or server where you are running ConsoleOne.

However, you should be aware of the following issues:

- ◆ [“Running ConsoleOne from a Workstation” on page 497](#)
- ◆ [“Running ConsoleOne from a Server” on page 497](#)

For information on managing software packages from multiple workstations, see [“What Are My Software Package Management Options?” on page 500](#).

Running ConsoleOne from a Workstation

If you run ConsoleOne from a workstation and save a software package to that workstation, the package will not be available in ConsoleOne to other workstations or servers running ConsoleOne.

Where you save software packages (on workstations or on servers) depends on how you want to manage the software packages.

Running ConsoleOne from a Server

You must have the same drive mapping to a server on different workstations if you run ConsoleOne from the server at those workstations. Otherwise, any software package you save to that server cannot be read at the different workstations.

For example, the following scenario illustrates when a package can be found:

1. You run ConsoleOne from Workstation A to access Server A.
2. Server A is mapped as drive S: for Workstation A.
3. You save PKG_A.SPK to Server A.
4. You run ConsoleOne from Workstation B to access Server A.
5. Server A is also mapped as drive S: for Workstation B.
6. PKG_A.SPK can be found because both workstations were mapped to drive S:.

The following scenario illustrates when a package cannot be found:

1. You run ConsoleOne from Workstation A to access Server A.
2. Server A is mapped as drive S: for Workstation A.
3. You save PKG_A.SPK to Server A.
4. You run ConsoleOne from Workstation B to access Server A.
5. Server A is mapped as drive T: for Workstation B.
6. PKG_A.SPK cannot be found because you are looking for the package on drive T: when it was previously saved to drive S:.

Note that the only difference between the scenarios is the drive letter mappings to Server A for each workstation.

Distributing Software Packages

Distributions can include software packages, which are installed, or File Groupings, which are extracted.

The Policy/Package Agent extracts or installs Software Package Distributions on the Subscriber server.

When software packages are created, they can contain system requirements that must be met before the package can be installed on the target Subscriber's server. If the Subscriber meets these requirements, the subscription schedule determines when the package will actually be installed.

Distributing Software Packages to a Cluster

When you send a Distribution containing software packages to a cluster to update the SYS: volume for each node, the only node in the cluster that will receive it is the one that currently has the Subscriber software running.

Because the machines comprising the nodes in the cluster run the Subscriber software, only one node at a time in a cluster will be actively running the Subscriber software.

Therefore, if you want to use a Software Package Distribution to update files on a SYS: volume for each node in a cluster, you must do this manually by updating one node, bringing it down so that the next node in the failover sequence will see that the previous node has failed and start running the Subscriber software, then update that machine, bring it down, and so on, until all of the machines in the cluster have been updated. Then restart all of the downed servers in the cluster and the primary node's machine will take over again.

You can use a Software Package Distribution to update files on the cluster machine itself, such as TED's .NCF files, because the Subscriber software will be contained on the cluster machine's shared hard drive.

Failure of Software Package Installations

If a server fails to meet any of the software package requirements, it is not installed.

Failure During an Installation

The system tracks all changes made by the installation of a software package. Except as noted under [“Rolling Back Software Package Installations” on page 498](#), if a server meets the requirements and the installation begins, then a failure condition halts the installation prematurely, the installation program will automatically return the server to the state it was in before the installation began, undoing what had been done to that point.

Failure of a Component

If a server meets the software package requirements, and some of the components meet the installation requirements met and some do not, the installation will be completed except for the components where the requirements were not met. In this case, you would have a partial installation of the package.

You should organize your software packages and their components so that if this happens, it will not leave disconnected or incomplete files or applications on the target machine.

Rolling Back Software Package Installations

You can undo a successful software package installation by rolling it back. However, any software package installation that runs a program such as a NetBasic script, a Java Class, or an NLM that modifies the server cannot be rolled back successfully.

Software package rollback is enabled by default. You should not disable rollback, unless you know the installation will never need to be undone.

Rollback Methods

There are two ways you can roll back a software package installation:

- ◆ Type **package rollback** at the server's ZfS console prompt for the server containing the package to be rolled back.
- ◆ Use a Web browser to access ZfS and select the rollback option. For more information, see [“Novell iManager” on page 359](#).

The software package will be uninstalled, leaving the server as if it had never been installed, except for any changes that might have been made to the server in using the installed application.

Rollback works, even if some components have not been installed during an incomplete package installation, because the installation program tracks what was and wasn't installed.

Rolling Back Previous Installations

When you roll back an installation, it will be the last software package installed on that server. If that's not the one you need to roll back, you must roll back more recent installations first.

For example, you installed three software packages on a server (Package1, Package2, and Package3). Package1 was installed first and Package3 was installed last. If you want to roll back Package2, you must first roll back Package3. To do so, you would type **package rollback** at the server's ZfS console prompt once for Package3, then again for Package2.

Planning Server Software Packages

Review each of the following sections and take notes as instructed. This information will help you to configure your software packages and their components.

- ◆ [“Which Files or Applications Do I Want to Distribute?” on page 499](#)
- ◆ [“What Are the Software Package Components?” on page 500](#)
- ◆ [“What Are the Minimum Requirements?” on page 500](#)
- ◆ [“What Are My Software Package Management Options?” on page 500](#)

Which Files or Applications Do I Want to Distribute?

You can distribute software packages containing files and applications for servers, as well as software packages containing end-user applications for further distribution in ZENworks for Desktops (ZfD) to workstations. For information on configuring a Desktop Application Distribution, see the [Novell Documentation Web site \(http://www.novell.com/documentation/lg/zdfs/index.html\)](http://www.novell.com/documentation/lg/zdfs/index.html), and under Administration, click Application Management.

If you have ZfD 4.0.1 installed, you can also distribute desktop applications using TED, instead of including them in software packages. For more information, see [Chapter 19, “Desktop Application Distribution,” on page 525](#).

You can include a file or application in more than one software package. For instance, a word processor application could be included in a software package designed for a secretarial group and one designed for a financial group.

Where applicable, organize the files and applications into logical groups for inclusion in software packages.

Follow the steps under [“Creating a Server Software Package” on page 509](#) and [“Creating the Software Package Components” on page 510](#) and note the information you will need to know for creating the software package and its components.

What Are the Software Package Components?

You can have one or more components in a software package. For example, if you create a software package for installing virus protection software, you might want one component to be the original virus protection program, and another component a current virus pattern update file.

Components in a software package can each have the same or different installation requirements. If you give the components different requirements, they might not all be installed together. You can save time and minimize error by giving all of the components the same requirements.

IMPORTANT: Files and applications that are dependent on each other should be included in the same component. This will prevent problems running the files or applications if a critical component is not installed. If you need to split an application's files into multiple components, make sure that you make each component's requirements the same, so that they will all install or not install together.

Follow the steps under [“Configuring the Software Package Components” on page 511](#) and note the information you will need to know for configuring the package components.

What Are the Minimum Requirements?

Minimum requirements establish whether a software package will be allowed to install on the target machine. If these requirements are all met, the software package can be installed on that server.

However, requirements can be established for the software package as a whole, as well as for each package component. Therefore, if the package's requirements were all met, but some component requirements were not met, only part of the package would be installed.

Follow the steps under [“Configuring the Server Software Package” on page 510](#) and note the information you will need to know for configuring the software package.

What Are My Software Package Management Options?

The following sections explain where to store Server Software Package files, and how to manage them:

- ◆ [“Understanding Server Software Package Files” on page 500](#)
- ◆ [“Understanding Your Software Package Management Options” on page 501](#)
- ◆ [“Storing and Managing .SPK Files Using One Workstation” on page 501](#)
- ◆ [“Storing .SPK Files on a Network Server and Managing Them from Multiple Workstations” on page 502](#)
- ◆ [“Example in Using a Master SNAPINPREFS.SER File” on page 504](#)

Understanding Server Software Package Files

There are three file types associated with software packages:

- ◆ **Configuration File (.SPK):** When you create a Server Software Package, you will initially create a configuration file (.SPK) for it. This file's configuration is created in the properties of the software package object in the Server Software Packages namespace in ConsoleOne.

.SPK files are generally small (around 100 KB). Therefore, they can generally be stored on the workstation running the instance of ConsoleOne that you are using to create and manage software packages.

- ◆ **Compiled File (.CPK):** When you compile a software package, a .CPK file is created from the .SPK file's configuration information. This provides the content of the software package, such as files or functions. The .CPK file is used to install the software package's content on a server.

You should generally store .CPK files on a server where there is sufficient free disk space, because compiled software packages may contain many files. However, small .CPK files that only contain functions can be stored on a workstation.

- ◆ **Preferences File (.SER):** The preferences file (SNAPINPREFS.SER) is automatically created on the workstation being used to create a software package. It contains pointers to the .SPK files for the software packages.

This preferences file allows you to see the software packages in the namespace in ConsoleOne. In other words, software packages will be displayed in the Server Software Packages namespace for an instance of ConsoleOne only if the .SPK file's path is listed in the preferences file located on the workstation running that instance of ConsoleOne.

When you create a new software package, you will specify the local path for the .SPK file. When you compile a software package, you will specify the server's path for the .CPK file. After you exit ConsoleOne, any time you have created, deleted, or compiled a software package, the .SPK file paths are logged to the SNAPINPREFS.SER file.

The path to the .CPK file is also logged to the SNAPINPREFS.SER file. Therefore, the next time you compile the software package, the wizard will be able to display the .CPK file's previous location so that you do not have to remember it each time you compile the package. However, you will need to note where you store the .CPK files for when you want to distribute them using TED, because the .CPK files' locations are not stored in the software package's properties.

Understanding Your Software Package Management Options

You have two options for managing software packages:

- ◆ Using One Workstation
- ◆ Using Multiple Workstations

If you will be using only one specific workstation for viewing, creating, and managing all of your software package files, then you can store the .SPK files on that workstation.

It is possible to manage your software packages from multiple workstations. This requires that you centralize your .SPK file storage to a network server. This method will also require the use of a master SNAPINPREFS.SER file so that you can view all of your software packages from any workstation.

The next sections explain these two options.

Storing and Managing .SPK Files Using One Workstation

If you will use only one workstation for viewing, creating, and managing your software packages, you can store the .SPK files on the workstation and the .CPK files on a server.

Whether you are running ConsoleOne from the workstation where it is installed or from a workstation that uses an installation of ConsoleOne on a network server, the SNAPINPREFS.SER file is updated on the workstation being used to run ConsoleOne.

Storing .SPK Files on a Network Server and Managing Them from Multiple Workstations

If you want to use multiple workstations for viewing, creating, and managing the same set of software packages, you will need to store all .SPK files on a network server so that they can be accessed by each workstation.

You may also want to use different workstations for managing different sets of software packages. Any workstation used to create .SPK files will have a software package preferences file of its own created on the workstation used to manage the software packages.

You can manage all of your software packages from multiple workstations if you use a master copy method for the SNAPINPREFS.SER file.

- ◆ [“Understanding the Software Package Preferences File” on page 502](#)
- ◆ [“Managing Software Packages from Multiple Workstations” on page 503](#)
- ◆ [“General Rules for Managing Software Packages from Multiple Workstations” on page 503](#)
- ◆ [“The Best Scenario for Using Multiple Workstations to Manage Software Packages” on page 504](#)

Understanding the Software Package Preferences File

When you create a Server Software Package object in ConsoleOne, a software package preferences file (SNAPINPREFS.SER) is created in the following location on the workstation running ConsoleOne:

`C:\Documents and Settings\user_ID\.consoleone` (Windows 2000)

or

`C:\Winnt\Profiles\user_ID\.consoleone` (Windows NT)

where *user_ID* is the user directory associated with how you are logged on, such as Administrator.

The full path and filename for a software package is drive-dependent. The SNAPINPREFS.SER file contains the drive letter, path, and package name for each .SPK created by the workstation.

The SNAPINPREFS.SER file is unique for each workstation. It is the preferences file that is updated whenever you add or remove .SPK files using that workstation. Therefore, if you use three different workstations to create .SPK files, you will have three different SNAPINPREFS.SER files, each on its own workstation.

When you start ConsoleOne, it checks to see if a SNAPINPREFS.SER file was created for that workstation by the instance of ConsoleOne being run on the workstation, and whether ConsoleOne is installed on that workstation or is being run on that workstation from an instance installed on a server. If the file does not exist, a SNAPINPREFS.SER file is created when you exit ConsoleOne. If it exists, the SNAPINPREFS.SER file is updated with the full paths to any new .SPK files.

You can copy a SNAPINPREFS.SER file from one workstation to another. However, after replacing a SNAPINPREFS.SER file with a copy from another workstation, you will need to restart ConsoleOne to see any change.

A software package can become unusable if you change drive mappings after creating the package, because the SNAPINPREFS.SER file's location to the package will then be different. However, if you use a UNC path, this is not an issue as long as the workstation has access to that UNC path.

If you replace the SNAPINPREFS.SER file on a workstation, you will need to manually insert any software packages missing from the newly copied SNAPINPREFS.SER file. Otherwise, the

software packages listed in the SNAPINPREFS.SER file that was replaced would be inaccessible on the workstation.

Even if a workstation has never been used to create a software package, you can copy a SNAPINPREFS.SER file from another workstation to the appropriate location (C:\...\CONSOLEONE). Then when you start ConsoleOne, you will see all of the software packages listed in the SNAPINPREFS.SER file that was copied.

For more information, see [“Example in Using a Master SNAPINPREFS.SER File” on page 504](#).

Managing Software Packages from Multiple Workstations

If you will be using multiple workstations for creating, deleting, and compiling the same set of software package files, you should do the following:

1. Store the .SPK files on one network server (usually the server where you are storing their corresponding .CPK files), so that the software packages can all be accessed from any workstation.
2. When mapping a workstation to the server where the .SPK and .CPK files are stored, use the same drive letter for all workstations.
3. Create a master SNAPINPREFS.SER file to use for keeping all workstations updated with their latest software package additions, deletions, and compilations (see [“Setting Up the Master SNAPINSPREFS.SER File” on page 506](#)).
4. Create a batch file for starting and stopping ConsoleOne on a workstation (see [“Creating the ConsoleOne Batch File” on page 507](#)). This batch file will do two things:
 - ◆ Automatically upload the latest SNAPINPREFS.SER file from the storage server to the workstation any time ConsoleOne is started on that workstation.
This will allow you to see all software packages from the workstation where you started ConsoleOne.
 - ◆ Automatically download the revised SNAPINPREFS.SER file from the workstation to the storage server when ConsoleOne is exited on that workstation.
This will create a new master copy of the .SER file containing the workstation’s latest software package additions.
5. Run the batch file from any workstation where you want to manage software packages (see [“Using the ConsoleOne Batch File” on page 509](#)).

General Rules for Managing Software Packages from Multiple Workstations

Using a master copy for the SNAPINPREFS.SER file will work only if you exit ConsoleOne on one workstation, then start it on another workstation. This sequential method will not work for concurrently running instances of ConsoleOne where each instance is updating its local SNAPINPREFS.SER file. The instance of ConsoleOne that is exited last will overwrite the master copy with its local .SER file.

IMPORTANT: Creating, deleting, or compiling software packages in ConsoleOne are the only functions that cause logging to the SNAPINPREFS.SER file. Therefore, you can use ConsoleOne to manage software packages, such as viewing and editing properties, without starting ConsoleOne from the batch file. Just make sure that you do not add, delete, or compile any .SPK files in ConsoleOne if you do not start ConsoleOne with the batch file.

To manage software packages using this master copy/single server/multiple workstation method, observe the following general rules:

- ◆ Always exit ConsoleOne after creating a new software package (.SPK file) or compiling a new .CPK file. This will cause the master SNAPINPREFS.SER file to contain the newest software package links.
- ◆ Never have two or more workstations concurrently managing software packages. The batch file used to start ConsoleOne on these workstations could cause paths to any newly created software packages to be lost.
- ◆ Never use the batch file to start ConsoleOne when you do not intend to manage software packages. Instead, start ConsoleOne without using the batch file.

You need to do this because the batch file will always overwrite the master copy on the software package storage server when ConsoleOne is exited (if ConsoleOne was started by the batch file). You could inadvertently overwrite the master SNAPINPREFS.SER file and lose links to newly created software packages.

For example, on Workstation_A you run the batch file to start ConsoleOne, do administrative work other than software packages, for some reason go to Workstation_B where you decide to create a new software package (so you use the batch file again), exit ConsoleOne on Workstation_B, then later exit ConsoleOne on Workstation_A. Your new software packages created on Workstation_B no longer have links to them in the master SNAPINPREFS.SER file.

The Best Scenario for Using Multiple Workstations to Manage Software Packages

The best scenario is that you have one administrator who can use multiple workstations to manage your software packages. Otherwise, if you have multiple administrators, they need to be coordinated so that they don't overwrite each other's latest software package additions and deletions in the master SNAPINPREFS.SER file.

For more information, see [“Example in Using a Master SNAPINPREFS.SER File” on page 504.](#)

Example in Using a Master SNAPINPREFS.SER File

Keeping the master copy on the server properly updated is a matter of timing. For example, in the following scenario, the first SNAPINPREFS.SER file was initially created on Workstation A, then copied down to the network server to be the master SNAPINPREFS.SER file. Both workstations are using Windows 2000.

A batch file is used to start ConsoleOne for the purpose of controlling events before and after using ConsoleOne.

1. Administrator A starts the batch file on Workstation A to begin ConsoleOne.
2. The batch file running on Workstation A identifies the storage server as being mapped to drive M: (or it maps drive M: to that server).
3. The batch file copies the master SNAPINPREFS.SER file from the server at drive M: to the C:\Documents and Settings\user_ID\CONSOLEONE directory on Workstation A.
4. Administrator A creates a new software package, naming it SSP1.SPK.
5. Administrator B starts the batch file on Workstation B to begin ConsoleOne.
6. The batch file running on Workstation B identifies the storage server as being mapped to drive M: (or it maps drive M: to that server).
7. The batch file copies the master SNAPINPREFS.SER file from the server at drive M: to the C:\Documents and Settings\user_ID\CONSOLEONE directory on Workstation B.

This is the same version of SNAPINPREFS.SER that Administrator A had copied up to Workstation A, except that it hasn't been updated yet with Administrator A's addition of SSP1.SPK.

8. Administrator B creates a new software package, naming it SSP2.SPK.
9. Administrator B exits ConsoleOne, which updates SNAPINPREFS.SER on Workstation B with the SSP2.SPK path.
10. The batch file running on Workstation B updates the master SNAPINPREFS.SER file on the network server at drive M: with the updated SNAPINPREFS.SER file from Workstation B.
This updated master SNAPINPREFS.SER file now contains the location of SSP2.SPK.
11. Administrator A exits ConsoleOne, which updates SNAPINPREFS.SER on Workstation A with the SSP1.SPK path.
12. The batch file running on Workstation A updates the master SNAPINPREFS.SER file on the network server at drive M: with the updated SNAPINPREFS.SER file from Workstation A.
This updated master SNAPINPREFS.SER file now contains the location of SSP1.SPK.
However, the location for SSP2.SPK has been lost, because Workstation B's update of the master SNAPINPREFS.SER file was overwritten by Workstation A's later update.

This scenario would cause Administrator B to lose access to SSP2.SPK, because the master SNAPINPREFS.SER file will no longer contain a record of SSP2.SPK's location. It was replaced with Administrator A's SNAPINPREFS.SER file containing only SSP1.SPK's location. However, SSP2.SPK can be manually inserted into ConsoleOne (using the Insert Software Package option), so that it will be listed in the SNAPINPREFS.SER file along with SSP1.SPK.

For this multiple-workstation management method to work, you must ensure that the master SNAPINPREFS.SER file you keep on the network server is only used by one workstation at a time for creating, deleting, or compiling .SPK files. However, you can use multiple workstations to simultaneously view or edit a Server Software Package object's properties, because the viewing and editing functions do not cause updates to a SNAPINPREFS.SER file.

WARNING: You can perform edits to the properties of the Server Software Package object without affecting the SNAPINPREFS.SER file. However, because Server Software Package objects are not in eDirectory™, but only in a name space, the .SPK files might not have file-locking protection, unless the server's operating system provides this functionality. Therefore, you should devise management controls to protect against overwriting .SPK files when using multiple workstations to manage software packages.

Setting Up Server Software Packages

To set up a software package for distribution, perform the following tasks in order:

1. [“Setting Up Multiple-Workstation Management for Server Software Packages” on page 506](#)
2. [“Creating a Server Software Package” on page 509](#)
3. [“Configuring the Server Software Package” on page 510](#)
4. [“Creating the Software Package Components” on page 510](#)
5. [“Configuring the Software Package Components” on page 511](#)
6. [“Compiling a Software Package” on page 521](#)
7. [“Distributing the Software Package” on page 521](#)

Setting Up Multiple-Workstation Management for Server Software Packages

If you want to manage your software packages from multiple workstations, do the following in order to set up managing the replication of a master copy of the SNAPINPREFS.SER to multiple workstations:

1. [“Setting Up the Master SNAPINPREFS.SER File” on page 506](#)
2. [“Creating the ConsoleOne Batch File” on page 507](#)
3. [“Using the ConsoleOne Batch File” on page 509](#)

Setting Up the Master SNAPINPREFS.SER File

For the following instructions, select any workstation that you will use for managing software packages. If you have already created software packages using a workstation, select that workstation so you will not lose any software package information stored in the workstation’s SNAPINPREFS.SER file.

- 1** Map a drive to the server where you want to store your .SPK and related .CPK files.

This drive letter should be one that can be used by all of the other workstations that you will use to manage software packages. This drive letter is written to the SNAPINPREFS.SER file as part of the path information for each listed .SPK file, so it should be a fixed drive letter that all workstations use.

The drive letter will also be used in the batch file that you use to start ConsoleOne, which will provide each workstation access to the same .SPK file locations.

- 2** If you already have Server Software Package objects created by this workstation, skip to [Step 5](#).

or

If you have not yet created any Server Software Package objects using this workstation, start ConsoleOne.

This version of ConsoleOne must have the Zfs Policy and Distribution services snap-ins installed.

- 3** In the Server Software Package namespace, create a Server Software Package object.

You do not need to fully configure the Server Software Package object at this time. Just give the package a name and provide a location and filename for the .SPK file. Make sure you use the drive mapping you used in [Step 1](#).

For information on creating software packages, see [“Setting Up Server Software Packages” on page 505](#).

- 4** Exit ConsoleOne.

This step is important to make sure that the SNAPINPREFS.SER file is created for this workstation.

- 5** On the network server you will use to store the master copy of the SNAPINPREFS.SER file, create a directory named C1 at the root of the drive.

You can select any safe location on the server for the master SNAPINPREFS.SER file.

The [batch file sample](#) provided below uses a directory named C1. You can modify the batch file if you want to use a different directory name, and you can include path information; however, do not use variables.

For example,

```
ZENWORKS\C1SSP
```

could be used to replace the C1 directory name.

6 Copy the workstation's SNAPINPREFS.SER file from:

```
C:\Documents and Settings\user_ID\.consoleone (Windows 2000)
```

or

```
C:\Winnt\Profiles\user_ID\.consoleone (Windows NT)
```

to the C1 directory on the network server.

This becomes the master SNAPINPREFS.SER file that will be updated with new .SPK paths, provided you are using the batch file documented in [“Creating the ConsoleOne Batch File” on page 507](#).

Creating the ConsoleOne Batch File

Review the following sections to create and use the batch file:

- ◆ [“Sample Batch File” on page 507](#)
- ◆ [“What the Batch File Does” on page 507](#)
- ◆ [“Creating Your Batch File” on page 508](#)
- ◆ [“Optional Modifications to the Batch File” on page 508](#)

Sample Batch File

```
@echo off
REM map a network drive
net use m: \\prv-ale.provo.novell.com\voll

REM create a backup copy of the workstation's .SER file
copy "%USERPROFILE%\.consoleone\snapinprefs.ser"
"%USERPROFILE%\.consoleone\snapinprefs.tmp"

REM copy the master .SER to the workstation
copy m:\c1\snapinprefs.ser "%USERPROFILE%\.consoleone\snapinprefs.ser"

REM start ConsoleOne
C:\Novell\ConsoleOne\1.2\bin\ConsoleOne.exe

REM batch file control returns after exiting ConsoleOne
REM copy the updated .SER to server
copy "%USERPROFILE%\.consoleone\snapinprefs.ser" m:\C1\snapinprefs.ser

REM restore the backup copy of the workstation's .SER file
copy "%USERPROFILE%\.consoleone\snapinprefs.tmp"
"%USERPROFILE%\.consoleone\snapinprefs.ser"

REM delete the mapped network drive
net use m: /delete
@echo on
```

What the Batch File Does

- ◆ It maps a network drive for accessing the server where you are storing .SPK and .CPK files.

- ◆ It uses the %USERPROFILE% Windows variable to locate the ZFS .CONSOLEONE directory. This variable is also used by ZFS to determine where it will create the .CONSOLEONE directory and write the SNAPINSPREFS.SER file.
- ◆ It creates a backup .TMP copy of the SNAPINSPREFS.SER file.
- ◆ It copies the master SNAPINSPREFS.SER file from the C1 directory on the server to the workstation's .CONSOLEONE directory.
- ◆ It starts ConsoleOne.
- ◆ After you have exited ConsoleOne, the batch file copies the updated SNAPINSPREFS.SER file from the workstation's .CONSOLEONE directory to replace the version in the C1 directory on the server. This becomes the new master SNAPINSPREFS.SER file.
- ◆ It restores the backed up copy of the SNAPINSPREFS.SER file from the .TMP file.
- ◆ It unmaps the drive letter to the server.

Creating Your Batch File

- ◆ Copy the text from the above sample batch file into a text editor.
- ◆ Replace the m: drive letter with one that each of your workstations has free. Make sure you do this wherever m: exists in the batch file.
- ◆ Edit the net use m: \\prv-ale.provo.novell.com\vol1 line by replacing it with the path to the server volume or shared folder of the server where you are storing the .SPK and .CPK files.
- ◆ Save the batch file on your workstation and give it a name, such as:

```
C1SSP.BAT
```

- ◆ Copy this batch file to each workstation that you will use to manage software packages.

Optional Modifications to the Batch File

- ◆ If you installed ConsoleOne to a different location on the workstation than is indicated in the batch file sample, modify the C:\Novell\ConsoleOne\1.2\bin\ConsoleOne.exe line to reflect the location of the CONSOLEONE.EXE file on the workstation.

You should make this modification in each individual batch file copy on a workstation where the default ConsoleOne path was not used.

- ◆ This batch file can also be used by a workstation to start an instance of ConsoleOne that is installed on a server. Modify the C:\Novell\ConsoleOne\1.2\bin\ConsoleOne.exe line to reflect the location of the CONSOLEONE.EXE file on the server. Make sure the drive letter is the one being used for accessing the server (see [Step 1 on page 506](#)).
- ◆ If the .CONSOLEONE directory path is different between workstations because the %USERPROFILE% variable was not used, you will need to edit any lines containing the variable, as necessary. Open the copy of the batch file on a workstation where the %USERPROFILE% variable was not used and edit the lines containing the variable to reflect the correct path to the .CONSOLEONE directory.
- ◆ If you created a directory other than C1 on the server, replace C1 wherever it exists in the batch file with the directory that you specified in [Step 5 on page 506](#).
- ◆ The batch file creates a .TMP version of the SNAPINSPREFS.SER file. This allows you to maintain the version of the .SER file on the workstation that existed before you used the batch

file. However, if you want the workstation's version to always match the master version it copied down to the server, remove the following two lines from the batch file:

```
copy "%USERPROFILE%\consoleone\snapinprefs.ser"  
"%USERPROFILE%\consoleone\snapinprefs.tmp"  
  
copy "%USERPROFILE%\consoleone\snapinprefs.tmp"  
"%USERPROFILE%\consoleone\snapinprefs.ser"
```

- ◆ If you cannot use the same drive letter for all workstations, you can use the %1 argument in the batch file, but only if you are using UNC paths for all of your .SPK files. To do this, replace all occurrences of m: with %1. Then, when you execute the batch file from a command line, add the drive letter after the batch file's name. For example,

CLASSP R:

will cause the batch file to use R: as the drive for locating the master copy of the SNAPINSPREFS.SER file.

Using the ConsoleOne Batch File

- ◆ Before running this batch file, place a SNAPINSPREFS.SER file in the ...\.CONSOLEONE directory of each workstation you will use to manage software packages. The batch file assumes that the .SER file will exist for copying and replacing.
- ◆ Before running this batch file, place your master copy of the SNAPINSPREFS.SER file in the C1 directory of the server where you have stored the software package files. The batch file assumes that this .SER file will exist for copying and replacing.
- ◆ Run this batch file any time you plan to add, delete, or compile software packages.
- ◆ You do not need to use the batch file when you view or edit the properties of software packages. The add, delete, and compile functions are the only actions that will cause the SNAPINSPREFS.SER file to be updated.

Creating a Server Software Package

To create software packages for distribution:

- 1** In ConsoleOne, right-click the Server Software Packages namespace > click New Package. The Create New Server Software Package Wizard opens.
- 2** Read the information on the first dialog box > click Next.
- 3** Enter a name for the software package.
Make this a descriptive name. It will be displayed in ConsoleOne under the Server Software Packages object.
WARNING: Do not use double-byte characters in the software package name. This will cause an error in any report you run on the software package.
- 4** Because software packages are file-based, enter the full path and filename, including the .SPK extension.

If you don't enter the extension, you will be prompted to add it.

You can also use UNC paths.

You can store the .SPK files on a workstation or server. The .SPK files is typically below 100 KB in size. However, compiled software packages (.CPK files) can be in the hundreds of

megabytes. For information on storing .SPK and .CPK files, see [“What Are My Software Package Management Options?” on page 500](#).

WARNING: Software package full paths and filenames are drive-dependent. A software package can become unusable if you change drive mappings after creating the package. Make sure your entry in this field will not change. However, if you used a UNC path, this is not an issue.

- 5 Click Finish.

Configuring the Server Software Package

Once a software package has been created, you need to configure it by setting the prerequisites for installation of the files and applications contained in the package.

To configure a package:

- 1 In ConsoleOne, right-click a software package > click Properties.

The Identification tab should be displayed. If not, click it.

The Name field should display the name you gave the package when you created it.

- 2 Enter a useful description for the software package.
- 3 If you don't want to be able to roll back to the older version of the server file or application after installing the newer version, check Disable Rollback. However, this is not recommended.

For information on rolling back software package installations, see [“Rolling Back Software Package Installations” on page 498](#).

- 4 Click the Requirements tab.

- 5 Click Add > select a requirement:

[“Operating System” on page 625](#)

[“Memory \(RAM\)” on page 626](#)

[“Disk Space” on page 626](#)

[“SET Commands” on page 627](#)

[“Registry” on page 627](#)

[“File” on page 628](#)

[“PRODUCTS.DAT” on page 628](#)

- 6 Repeat [Step 5](#) for each requirement.

- 7 If you want to use variables to customize the installation, click the Variables tab > Add.

- 8 Enter the variable name and value.

For information on variables, see [“Using Variables to Control File Extraction” on page 584](#).

- 9 Repeat [Step 7](#) and [Step 8](#) for each variable.

- 10 Click OK when you have finished configuring.

If you click Cancel, none of the configuration changes on any of the tabs will be saved.

Creating the Software Package Components

Once you have created and configured a software package, you need to create the components of the package, which includes the individual files or applications for the package.

To create the software package components:

- 1** In ConsoleOne, right-click a software package (in the left pane) > click New Component.
- 2** Enter the name of the component as you want it to be displayed in ConsoleOne > click OK.
The component is displayed as named under the Software Package object.
- 3** Repeat these steps for each component needed.

Configuring the Software Package Components

Once you have created the software package components, you need to configure the prerequisites for each, including identifying the files or applications for the component.

Package components can each have the same prerequisites, which can save time and minimize user error.

To configure a component:

- 1** In ConsoleOne, right-click a component > Properties.
The Identification tab should be displayed. If not, click it.
- 2** Enter a useful description for the component.
- 3** To determine what should happen after the package has been installed, select an option from the After Package Installation Is Complete drop-down list.
- 4** To continue configuring the component, see each of the following that you might need to configure:
 - “Requirements” on page 511
 - “Pre-Installation Load/Unload Order” on page 512
 - “Pre-Installation Scripts” on page 512
 - “Copy File” on page 514
 - “Text File Changes” on page 517
 - “SET Commands” on page 518
 - “Registry Settings” on page 518
 - “PRODUCTS.DAT” on page 519
 - “Post-Installation Unload/Load Order” on page 520
 - “Post-Installation Scripts” on page 520
- 5** Click OK.
If you click Cancel, none of the configuration changes on any of the tabs will be saved.
- 6** Continue with “[Compiling a Software Package](#)” on page 521 to ready your software package for distribution.

Requirements

To specify requirements for installing the server files or applications:

- 1** While displaying the properties of the software package component, click the Requirements tab > click Add.
- 2** Select any of the following requirement items:
 - “Operating System” on page 625

- “Memory (RAM)” on page 626
- “Disk Space” on page 626
- “SET Commands” on page 627
- “Registry” on page 627
- “File” on page 628
- “PRODUCTS.DAT” on page 628

For further instructions on configuring an item, see one of the above items.

Pre-Installation Load/Unload Order

To configure certain NLM files or processes to load or unload before installing the software package on a server:

- 1** While displaying the properties of the software package component, click the Pre-Installation tab > click Load/Unload.
- 2** Click Add.
- 3** Select one of the following:
 - “Load NLM/Process” on page 623
 - “Load Java Class” on page 623
 - “Unload Process” on page 624
 - “Start Service” on page 624
 - “Stop Service” on page 624

For further instructions on configuring an item, see one of the above items.

IMPORTANT: If you select a process to be loaded by the software package, and it is already running on the target server, the package installation will fail and will be rolled back (if rollback is enabled). If the process requires intervention to unload, you must remember to unload it manually before installing the software package.

To make sure that a process is not already loaded when you are including it in the software package, add an unload option for that process before adding the load option—but only if the process does not require user input from the keyboard to unload it.

- 4** Repeat **Step 1** through **Step 3** for each NLM or process to be included.
- 5** Use the arrow keys to arrange the order to execute the NLM files and the processes.

Do not click OK until you have finished configuring the other tabs.

Pre-Installation Scripts

To configure running server scripts before installing the software package on a server:

- 1** While displaying the properties of the software package component, click the Pre-Installation tab > Script.
- 2** Click Add.
- 3** Enter the script name.
- 4** Select the script type (NCF, NetBasic, PERL).
- 5** Enter the script text.

WARNING: If a software package passes all requirements and executes the script, processing done by the script cannot be undone by rollback.

- 6** Repeat **Step 2** through **Step 5** for each script to be added.
- 7** Use the arrow keys to arrange the order to execute the scripts.

Do not click OK until you have finished configuring the other tabs.

Local File Copy

The Local File Copy component enables copying of files on a server from one location to another using a software package. You can either copy or move the files.

To configure the Local File Copy component:

- 1** While displaying the properties of the software package component, click the Local File Copy tab.
- 2** Click Add.

Local File Copy #1 defaults. You can edit that name.

- 3** Fill in the fields:

Source Path: Enter the full path where the files to be copied are located.

You can use wildcards in the path:

* = any number of characters

? = any single character in that position

??? = any characters in those positions

Target Path: Enter the full path where the copied files are to be placed.

You can use wildcards in this path. This path does not need to mirror the source path. However, you could mirror an existing target path.

Include Subdirectories: Includes all subdirectories and their files beginning from the directory at the end of the path; otherwise, only the files in the directory at the end of the path will be copied.

Maintain Attributes: Maintains the file attributes in the target's file system that exist in the source's file system.

Overwrite Destination Files: Overwrites files of the same name in the destination directories, regardless of differences in file dates. If you do not enable this option, files of the same name will not be replaced.

Maintain Trustees: Maintains the file's trustee attributes.

When a File Is Locked: Select one or both:

- ◆ **Retry __ Times:** Retries overwriting a locked file the number of times you select before failing to replace the file. Leave this check box unchecked to not replace locked files on the target file system.
- ◆ **Kill Connection of Open Files:** Attempts to kill the connection of locked files so they can be overwritten. This applies only to files being extracted, not to files being accessed to build the Distribution. If a file belonging to a Distribution is locked when the Distribution is being built, the build will fail. Server and NLM connections cannot be killed.

Error Processing: Fail On Error is checked by default. This stops the file copying process when an error is encountered in copying. To continue file copying when an error is encountered, click Continue On Error.

Operation: Sets whether to copy or move the files identified in the Source Path.

Do not click OK until you have finished configuring the other tabs.

Copy File

You can configure the Copy File component to control how files are copied during installation of a software package. This includes adding files to existing directories, creating new directories, adding files and subdirectories to the new directories, and deleting existing files and directories.

File Group is a root item for the component's expandable tree structure. You can have multiple File Groups for the Copy File component. A File Group is a set of related directories and files. File Groups are top-level items and cannot contain other File Groups.

The other structure items are Directory and File, which are contained within a File Group. Directories can contain other directories or files, but not File Groups.

IMPORTANT: When you add a File Group or Directory, you are creating the target paths where the files will be copied, not the source paths of the files. The source paths are automatically accounted for as you select your source files or directories.

To configure copying files during installation of the software package:

1 While displaying the properties of the software package component, click the Copy File tab.

2 To create your first File Group, do the following:

2a Click the down arrow on the drop-down box next to the Add button > select Add File Group > click Add.

Because files and directories must be contained within File Groups, you will be prompted to create a File Group the first time you click Add, regardless of which type you are attempting to add.

You should create one File Group for each specific target location. For example, C:\FILES, C:\DATA\ACCOUNTING, and C:\DATA\PERSONNEL could be different locations on a C: drive where you want to copy different groups of unrelated files.

2b Name the File Group > enter its target path.

The File Group's target path specifies the base path from where all Directories and Files within the group will be installed.

2c To specify what to do when a File Group location is locked, click the check box for one of the following:

1. Retry (enter the number of retry times)
2. Kill Connection of Open Files
3. Fail With Error

Retries are about 5 seconds apart. Therefore, 12 retries would take about one minute.

3 To create a target directory under a File Group or another directory, select the file group or directory > in the drop-down box, select Add Directory > click Add > do the following:

3a Because Directory is the default directory name, to rename the directory, right-click Directory > click Rename > enter the desired directory name > press Enter.

You must press Enter for the name change to be made.

To match an existing target directory for deleting or copying files, you must enter the exact name.

IMPORTANT: If you enter an existing directory name and that directory is marked as READ ONLY on the destination server's file system, the Software Package Distribution will fail when the Subscriber tries to extract the Distribution, because it will not be able to write to that directory. Therefore, you must know the attributes of existing target directories and remove their READ ONLY directory attributes.

You must create the same directory structure in the File Copy component as exists in the target location so that the directory name you enter here will be in the same sequence in the path.

- 3b** To determine whether to create or delete the directory, select the mode from the Copy Mode drop-down list.

Create: If you select Create and the directory does not exist, the directory will be created. If you select Create and the directory does exist, the directory will not need to be created, and no error will be encountered.

Delete: If you select Delete and the directory exists, the directory will be deleted, including any subdirectories and files under it. If you select Delete and the directory does not exist, the directory will not need to be deleted, and no error will be encountered.

WARNING: If you plan to set the Copy Mode as Delete for any directories you add, and you do not want any parent directories that you have added to also be deleted, place those parent directories in the Target Path field of the File Group. For example, if you want to delete C:\WINNT\COOKIES, but do not want to delete the WINNT directory, enter C:\WINNT in the Target Path field > click Add to enter the COOKIES directory in the tree structure > click Delete for the Copy Mode field. For example:

Target = C:\WINNT

Tree structure = COOKIES

causes only COOKIES and all of its files and subdirectories to be deleted.

Conversely, both the WINNT and COOKIES directories will be deleted if you enter C:\ in the Target path field > click Add to enter the WINNT directory in the tree structure > click Add to enter the COOKIES directory under WINNT in the tree structure > click Delete for the Copy Mode field. For example:

Target = C:\

Tree structure = WINNT\COOKIES

causes WINNT and all of its files and subdirectories to be deleted.

- 4** To add files or source directories under a file group or directory in the tree structure, select a file group or directory > in the drop-down box, select Add File > click Add > do the following:

- 4a** Select the files or directories using the Open dialog box.

These directories and files are displayed directly under the file group or directory you selected in [Step 3](#).

For the destination server's file system, attributes of the copied files and directories are not maintained. For more information, see [Step 4c](#).

If you selected a directory on the Open dialog box, it will not be displayed expanded. Click the plus signs to expand the existing structure under the directory that you added.

In the Open dialog box, the Recurse Directories option is checked by default. To only select files in this directory, click the Recurse Directories check box to disable it and none of the subdirectories will be selected.

To exclude files or subdirectories from being selected, click the Exclude Selected Subdirectory option > click the files or directories to be excluded (use Shift and Ctrl for multiple select) > click Open.

If you exclude files or subdirectories, it does not remove them from the file system. It only prevents them from being selected.

For information on removing files or subdirectories from the tree structure after adding files and directories, see [Step 8](#).

4b To configure a subdirectory that was added, do the following:

- ◆ Click the subdirectory > select the Copy Mode (whether to Create or Delete the directory).

WARNING: When you set the Copy Mode to Delete, it will cause deletion of the target directory and all of its files and subdirectories.

- ◆ To rename a subdirectory that was added, right-click the subdirectory > click Rename > enter a new directory name > press Enter.

You must press Enter for the name change to be made.

If you rename a directory that was selected through the Open dialog box, make sure that the new name meets your expectations for the target location.

Because only selected files have their path remembered for copying, renaming a directory does not affect file selection. In other words, you can give a target directory a different name than its source, and still have the same files copied under it.

4c To configure an added file, click the file > do the following:

- ◆ To determine the file's copy mode, select a mode from the Copy Mode drop-down list.

You must select an option for every file. You can select multiple files where you want the mode to be the same.

The options are: Copy Always, Copy If Exists, Copy If Does Not Exist, Copy If Newer, Copy If Newer and Exists, and Delete.

WARNING: When you set the Copy Mode to Delete, it will cause deletion of the selected file from the target server.

- ◆ Click the check box for each attribute that should apply to the selected files.

Attributes do not default. You must set them for the destination server. They are not carried over from where you obtained the file.

IMPORTANT: Do not check all of the attributes for a file, or an exception will be thrown on the server.

WARNING: When setting the attribute of an executable file, set it to Read Only. Do not set it to Execute. If you mark a file as Execute, the NetWare® CLIB API does not allow you to change it to a different attribute. To change the attribute from Execute to Read Only once the software package has been installed, you would need to manually delete the file, replace it, then set its attribute again.

5 To create another File Group, do the following:

- 5a** Click the down arrow on the drop-down box next to the Add button > select Add File Group > click Add.

It doesn't matter what you have selected in the tree structure, the File Group is automatically placed at the first tree level, equal to any other File Groups that are displayed.

5b Name the group.

5c Enter its target base path.

5d To indicate what to do when a group location is locked, click the check box for one of the following:

1. Retry (enter the number of retry times)
2. Kill Connection of Open Files
3. Fail With Error

6 Repeat **Step 5** through **Step 5d** for each additional File, Directory, or File Group to be added.

7 If you want the File Groups to be copied in a particular order, use the arrow keys to arrange the order of the File Groups.

The arrows will be dimmed if the File Group you have selected has no valid up or down movement available to it.

8 To remove a File Group, Directory, or File, select it > click Remove.

You can use the Remove button to prune the tree structure of unwanted files or directories.

You can use the Shift and Ctrl keys to select multiple items for removal.

IMPORTANT: If you remove a File Group or Directory, all Files and Directories displayed below it are also removed, but only from this tree structure, not from the source file system.

Do not click OK until you have finished configuring the other tabs.

Text File Changes

To configure making changes to text files during installation of the software package:

1 While displaying the properties of the software package component, click the Text Files tab.

2 Click Add.

After one text file has been added, you will be given the opportunity to select whether you are adding another text file or another change item for the selected text file.

To add another text file: Select Text File. It does not matter which text file or change item is selected in the left pane—the text file will be added to the far left level.

To add another change to a text file: In the left pane click the text file for the change > click Add > select Change. The change item will be added under the selected text file.

3 If you are adding a text file, enter the name of the text file.

4 Accept the default name (such as Change #1) or rename it.

If you are adding a text file, click OK.

5 Click the down-arrow for the Change Mode field > select the change mode from the drop-down list.

6 Click the down-arrow for the Search Type field > select the search type from the drop-down list.

7 Enter the exact search string.

8 Check the box if you want the string search to be case sensitive.

- 9** To find all occurrences of the search string, select the box (default); otherwise, deselect the box to find only the first occurrence.
- 10** Click the down-arrow for the Result Action field > from the drop-down list, select the action that should result if a string is matched.
- 11** If you will be replacing a string or entering a new one, enter the text in the New String text box.
- 12** Repeat **Step 2** through **Step 11** for each text file to add or each change to be made.
- 13** To reorder the text files and change items, use the arrow keys.

Do not click OK until you have finished configuring the other tabs.

SET Commands

For NetWare only.

To configure the target server's SET commands:

- 1** While displaying the properties of the software package component, click the Set Commands tab.
- 2** Click Add to open the NetWare Server SET Commands Wizard.
- 3** Select the server containing the SET commands > click Next.

IMPORTANT: The Policy/Package Agent, the ZWS Agent, and Java must be running on the server where you want to obtain the SET commands.

- 4** Select all of the SET commands you want to configure for the target server.

You can select whole categories by clicking the check box for the category, or click the plus sign to expand a SET command category and click the check boxes for individual SET commands to be included.

WARNING: Do not select the Set Developer Option SET command and change Off to On. This parameter is meant to help developers debug server abends. It disables some operating system checking to prevent certain abends from occurring. Also, if the Set Developer Option is turned on, running NCP™ scripts that require keyboard entry could abend the server.

- 5** Click Finish when you have completed selecting SET commands.

The selected SET commands are now displayed in the Set Commands tab for the file or application component.

- 6** To edit a SET command, click its plus sign to expand its attributes.
- 7** To edit an attribute, click the attribute > Edit.

A dialog box is displayed where you can make changes to the attribute.

- 8** Repeat **Step 7** for each attribute to edit for a given SET command.
- 9** Repeat **Step 6** through **Step 8** to edit another SET command's attributes.

Do not click OK until you have finished configuring the other tabs.

Registry Settings

To configure registry changes for either NetWare or Windows servers:

- 1** While displaying the properties of the software package component, click the Registry Settings tab > click HKEY_LOCAL_MACHINE.

HKEY_LOCAL_MACHINE is a Windows registry key. For NetWare, HKEY_LOCAL_MACHINE is also recognized by ZfS as the equivalent to My Server. Therefore, you can use this key for editing both NetWare and Windows registries.

- 2** Click Add.
- 3** Select from the following:
 - “Key” on page 629
 - “Binary” on page 630
 - “Expand String” on page 630
 - “(Default)” on page 630
 - “DWord” on page 631
 - “Multi-Value String” on page 631
 - “String” on page 631

For further instructions on configuring an item, see one of the above items.

- 4** Repeat **Step 2** and **Step 3** for each registry entry to be made.
- 5** Use the arrow keys to arrange the order in making registry entries.

Do not click OK until you have finished configuring the other tabs.

PRODUCTS.DAT

For NetWare only.

The PRODUCTS.DAT file can be updated by your software package so that future updates can know the most recently installed version of the file or application.

WARNING: Modifying PRODUCTS.DAT could prevent something from running or being installed on the NetWare server. Never modify any entries supplied by Novell.

To determine which action to take for PRODUCTS.DAT:

- 1** While displaying the properties of the software package component, click the Products.dat tab.
- 2** Select one of the following:

Option	Description
Add	Adds a new entry
Modify Existing Entry	Searches for a matching ID and modifies the version and description
Replace Existing Entry	Searches for a specific ID and replaces it with a new one
No Action	This is the default. Nothing is done to PRODUCTS.DAT

- 3** If you selected Add:
 - 3a** Enter the ID of the item to add.

This is case sensitive. The item is the ID of the new product for the .DAT file.
 - 3b** Enter the exact version number to add.
 - 3c** Enter the description to add.

- 4** If you selected Modify Existing Entry:
 - 4a** Enter the ID of the item to search for (case sensitive).
 - 4b** Enter the new version number.
 - 4c** Enter the new description.
- 5** If you selected Replace Existing Entry:
 - 5a** Enter the ID of the item to search for (case sensitive).
 - 5b** Enter the exact version number to match.
 - 5c** Enter the new ID.
 - 5d** Enter the new version.
 - 5e** Enter the new description.

Do not click OK until you have finished configuring the other tabs.

Post-Installation Unload/Load Order

To configure certain NLM files and processes to load or unload after installing the software package on a server:

- 1** While displaying the properties of the software package component, click the Post-Installation tab > click Load/Unload.
- 2** Click Add.
- 3** Select one of the following:
 - “Load NLM/Process” on page 623
 - “Load Java Class” on page 623
 - “Unload Process” on page 624
 - “Start Service” on page 624
 - “Stop Service” on page 624Click an item for further instructions on configuring it.
- 4** Repeat **Step 2** and **Step 3** for each NLM or process to be included.

Do not click OK until you have finished configuring the other tabs.

Post-Installation Scripts

To configure running NetWare server scripts after installing the software package on a server:

- 1** While displaying the properties of the software package component, click the Post-Installation tab > click Script.
- 2** Click Add.
- 3** Enter the script name.
- 4** Select the script type (NCF, NetBasic, PERL).
- 5** Enter the script text.

WARNING: If a software package passes all requirements and executes the script, processing done by the script cannot be undone by rollback.
- 6** Repeat **Step 2** through **Step 5** for each script to be added.

7 Use the arrow keys to arrange the order to execute the scripts.

Do not click OK until you have finished configuring the other tabs.

Compiling a Software Package

Your software packages (.SPK files) cannot be installed by Policy and Distribution Services until they have been compiled and have the .CPK extension.

To compile a software package:

1 In ConsoleOne, right-click a software package > click Compile Package.

The Compile Server Software Package Wizard opens.

2 Read the information on the first dialog box > click Next.

3 Enter a name and path for the compiled software package (using the .CPK extension) > click Next.

Select a location where free disk space is adequate for the .CPK file. Compiled software packages (.CPK files) are generally much larger than the uncompiled (.SPK) counterparts.

IMPORTANT: If you enter the path and filename of the .SPK when prompted for the compiled (.CPK) filename, the .SPK will be overwritten and can no longer be edited. Therefore, be sure to use the .CPK extension when naming the compiled version.

The compiling process could take some time, depending on how many files are involved.

4 When compiling has completed, click Finish.

5 Continue with “[Distributing the Software Package](#)” on page 521 to distribute your software package (.CPK).

Distributing the Software Package

Once a software package is ready for distribution, you can distribute it in the following ways:

- ♦ Manually copy the software package file (.CPK) to the server and run it from the server’s console prompt using the PACKAGE command (see [Appendix B, “Server Console Commands,”](#) on page 617 for instructions on using the command)
- ♦ Use TED (see [Chapter 16, “Tiered Electronic Distribution,”](#) on page 371 for instructions on distributing through TED)

Once a software package is installed on a target server, you might need to reboot the server. For example, if TCPIP.NLM is modified by the package, it cannot be downed—you must instead reboot the server to run that NLM again. However, you could have the software package cause the server to come down and restart automatically.

Converting Older Server Software Packages to ZfS 3.0.2

ZfS provides a wizard for converting older Server Software Packages to ZfS 3.0.2. The conversion works for both ZfS 1.0 and ZfS 2 software packages.

This wizard does not ship on the *ZENworks for Servers Program* CD or the *ZENworks 6 Server Management Program* CD. To obtain the wizard, search the Knowledgebase at [Novell Technical ServicesSM \(http://support.novell.com\)](http://support.novell.com) for TID 2962260. Instructions for installing the wizard are included in the Readme file.

After installing the wizard, to convert older .SPK or .CPK files to ZfS 3.0.2:

- 1** Make sure you have Write rights to any location where will be placing the converted versions of the software package files.
- 2** In ConsoleOne, do one of the following:
 - ◆ Right-click the Server Software Package namespace > click Convert Software Packages to Version 3.
 - ◆ Click the Server Software Packages namespace > File > Actions > Convert Software Package to Version 3.

The Convert Server Software Package Wizard starts.

- 3** On the Server Software Packages to Convert page, click Add > browse for and select the .SPK and .CPK files.

The wizard automatically checks the software package versions. If you selected ZfS 3.0.2 files, a message will be displayed indicating the files are already version 3.0.2.

This page will also not display older files that cannot be converted (for example, the file is corrupted, or a different file type was renamed to the .SPK or .CPK extension).

For the older files that you selected, the wizard page displays the original filenames under the Old Software Packages column. The New Software Packages column lists the new filenames, which are created by simply inserting `_v3` before the file extension.

You can resize the columns to read the full filenames and paths.

- 4** To change the new software package name or path, click a filename under either column > click Edit > edit the name or path > click OK.

You can change only the newer software package filename and path. However, the person running this wizard must have Write rights to any locations displayed under the New Software Packages column.

WARNING: You are allowed to rename the newer file to the same name as the older file. If you do so, the older file will be replaced with the converted file. If you click Cancel on the next wizard page, all converted files are deleted. Therefore, you could lose your original version of an older software package file. When you rename the newer file to be the same as the older file, you will receive a warning message asking if you are sure.

- 5** To remove a file (.SPK or .CPK) from being converted, click the filename under the Old Software Packages column > click Delete.

This only removes the file from the list, not from the hard drive location.

- 6** When finished selecting files, to start the conversion process, click Convert.

A message is displayed during the conversion process.

The converted files are located where you specified under the New Software Packages column. When you click Close to exit the wizard on the next wizard page, the files will be left at that location for your use.

After the files have all been converted, the Converted Server Software Packages page is displayed.

The Status column displays the conversion status of each listed file. It will indicate any files that could not be converted.

- 7** To automatically add the converted software packages to the Server Software Packages namespace in ConsoleOne, on the Converted Server Software Packages page, click the Add the Above Software Packages to ConsoleOne check box.

This option can save you from having to manually add each new package to ConsoleOne.

- 8** Click Close to exit the wizard.

If you clicked the Add the Above Software Packages to ConsoleOne check box, the converted software packages will be automatically added to the Server Software Packages namespace in ConsoleOne.

WARNING: If you click Cancel, the converted software package files will be deleted from the hard drive location that you specified under the New Software Packages column on the Server Software Packages to Convert page. If you renamed a newer file to be the same name as the older file, you will lose that older file when you click Cancel.

The converted software packages can now be used by ZfS 3.0.2 servers.

19 Desktop Application Distribution

Novell® ZENworks® for Servers (ZfS) 3.0.2 provides Policy and Distribution Services integration with ZENworks for Desktops (ZfD) 4.0.1 Novell Application Management.

For information on interoperability issues between ZfS 3.0.2 and ZfD 4.0.1, see [“Interoperability Between ZENworks for Servers and ZENworks for Desktops”](#) in the *Installation* guide.

The following sections provide information on understanding, setting up, and using the integration between ZfS and ZfD:

- ◆ [“Understanding Desktop Application Distributions”](#) on page 525
- ◆ [“Requirements”](#) on page 538
- ◆ [“Creating a Desktop Application Distribution”](#) on page 539
- ◆ [“Rebuilding Desktop Application Distributions”](#) on page 545
- ◆ [“Cleaning Up Desktop Application Distribution Files”](#) on page 546
- ◆ [“Sending Desktop Application Distributions Tree-To-Tree”](#) on page 547

Understanding Desktop Application Distributions

ZfS allows you to solve geographic, workload, and redundancy issues for applications distributed by Novell Application Launcher™ that might otherwise require much of your time in manual configuration work in ZfD. Review the following sections to see how ZfS can help you to automate much of your desktop application work.

- ◆ [“The Purpose of Desktop Application Distributions”](#) on page 525
- ◆ [“Distributed Application Issues”](#) on page 527
- ◆ [“Miscellaneous Issues”](#) on page 537

The Purpose of Desktop Application Distributions

- ◆ [“Applications in ZfD”](#) on page 525
- ◆ [“Distributed Applications in ZfS”](#) on page 526

Applications in ZfD

In ZfD, you can create Application objects so that users or workstations can receive their applications through Novell Application Launcher. An Application object contains pointers to the files belonging to the application, and also contains configuration parameters for how the application is to be installed and configured on the desktop.

In ZfD, the files belonging to an application can exist on any server, and the related Application object can exist anywhere in the tree. Therefore, for a workstation to receive an application through

Novell Application Launcher, the application's files are copied from a server and installed on the workstation.

However, problems can arise for the ZfD administrator, such as:

- ◆ **Network Traffic:** Many users or workstations can create heavy network traffic (especially across slower WAN links) to obtain their applications

To address the geographic issue of heavy network traffic, if you use only ZfD, you would need to do a lot of manual work. You would have to re-create and custom-configure the Application objects multiple times and copy their files to the various servers that would locally service their workstations.

- ◆ **Local Application Access:** Users need local access to their applications no matter where they connect to their network

You must manually create duplicate Application objects and create a site list in each copy of the Application object.

For more information on site lists, see "Setting Up Site Lists" in the *ZENworks for Desktops Administration* guide in the [ZENworks for Desktops 4 documentation Web site \(http://www.novell.com/documentation/lg/zdpr/index.html\)](http://www.novell.com/documentation/lg/zdpr/index.html).

- ◆ **Server Overload:** A server loaded with various application files can be over-worked to service all of its workstations

If you use only ZfD, you can configure Load Balancing (sharing the distribution workload between servers) in an Application object to address a server overload condition by having multiple servers being able to perform the same service. However, you would need to do a lot of manual work to use this feature.

- ◆ **Server Redundancy:** If a server loaded with various application files goes down, its workstations cannot receive those applications

If you use only ZfD, you can configure Fault Tolerance (server redundancy) in an Application object to address the situation where a server goes down by having multiple backup servers listed in the Application object. However, you would need to do a lot of manual work to use this feature.

ZfS provides solutions to resolve these geographic and manual work issues. To see how, continue with "[Distributed Applications in ZfS](#)" on page 526.

Distributed Applications in ZfS

ZfS provides a Desktop Application Distribution that allows you to minimize your network traffic, local application access, server bandwidth, and redundancy issues with less effort on your part.

For example:

- ◆ **Network Traffic:** Create a Desktop Application Distribution that contains your applications, then the Subscribers in each of your geographic areas create local copies of these applications. There, Novell Application Launcher can use these local applications to service the Subscriber server's users and workstations.
- ◆ **Local Application Access:** After creating and sending a Desktop Application Distribution, link up site lists so that users who travel between geographic locations can have local access to their applications.

For information on how Server Management sets up site lists, see [Step 13 on page 544](#).

- ◆ **Server Overload:** Through the Load Balancing feature, you can utilize multiple servers to service a large number of users or workstations via Novell Application Launcher. Simply use a common working context for each of the servers receiving the Desktop Application Distribution. Then, you have multiple servers available for load balancing.

IMPORTANT: Load balancing is concerned with access to the source paths on Subscriber servers, not with access to the distributed Application objects.

- ◆ **Server Redundancy:** Through the Fault Tolerance feature, you can have redundancy when servers go down by having other servers equally able to service your users and workstations via Novell Application Launcher. Simply use a common working context for each of the servers receiving the Desktop Application Distribution. Then, you have multiple servers available for fault tolerance.

IMPORTANT: Fault tolerance is concerned with access to the source paths on Subscriber servers, not with access to the distributed Application objects.

To do these things, you simply need to:

1. Create one Desktop Application Distribution for an application, or group of applications.
2. Send the Distribution to multiple servers.
ZfS automatically configures the application according to each server's environment.
3. Manually assign the necessary users or workstations to the groups that are associated with the new Application objects.
4. Click one button to link up the site lists.

Each server then has:

- ◆ Its own copy of an application's files on its file system
- ◆ Access to the Application object pointing to those files

The Application object is used to install the application on the workstations through Novell Application Launcher.

The Distribution process automatically does the multiple Application object creation, custom configuration, and file-copying work.

You can also send Desktop Application Distributions to other trees.

To further understand how ZfS can resolve these issues, continue with [“Distributed Application Issues” on page 527](#).

Distributed Application Issues

When sending a Desktop Application Distribution, some content in an Application object is kept, some is not kept, and some is modified. The following sections explain this:

- ◆ [“Understanding Golden and Distributed Application Objects” on page 528](#)
- ◆ [“Maintaining a Golden Application's Attributes” on page 528](#)
- ◆ [“Maintaining Associations When Distributing Objects” on page 530](#)
- ◆ [“Maintaining Application File Rights” on page 531](#)
- ◆ [“Subscriber Working Context Conflicts” on page 532](#)
- ◆ [“Maintaining Source Paths” on page 532](#)

Understanding Golden and Distributed Application Objects

When you create a Desktop Application Distribution, you select an application object to be distributed. In ZfS, this is known as the “golden” Application object. All of the Application objects that are created by the Distribution are referred to as the “distributed” Application objects.

Uniqueness of Golden Applications

As an administrator, you should keep track of which objects are golden Application objects for the Distributions, because Application objects themselves do not have any visual designation in ConsoleOne® to identify them as such.

Because normal ZfD activity associated with Application objects can cause the object’s internal revision number to change, unnecessary deltas of a Distribution could be triggered and sent. For example, a Distribution rebuild could be triggered by a simple change in a User Group object that is associated with an application contained within the Distribution, which information is not even transferred to the distributed applications. Therefore, your golden Applications should not be used by users or workstations.

We recommend that you keep your golden Application objects in a unique Novell eDirectory™ context and associate users and workstations to only the distributed Application objects. For more information, see [“Rebuilding Desktop Application Distributions” on page 545](#).

Synchronizing Golden and Distributed Applications

When a Distribution is rebuilt and resent, all distributed Application objects are synchronized with the golden Application object. In other words, if you make important changes in a distributed Application object, but not the golden Application object, then you rebuild and send the Distribution again, you could lose your changes, because the Distribution only uses the content in the golden Application object to update the distributed objects. Therefore, the golden Application objects are the only objects that you should modify when you want to re-send the Distribution.

However, you can make changes to distributed Application objects that will not be overwritten, if those changes are in the attributes that are not normally overwritten by a re-sent Distribution. This is explained in the next section.

Continue with [“Maintaining a Golden Application’s Attributes” on page 528](#).

Maintaining a Golden Application’s Attributes

ZfS distributes most attributes that exist in a golden Application object, but not all of them. Therefore, various outcomes can occur for the attributes contained in distributed Application objects any time a Distribution is rebuilt.

The following sections provide information on when attributes are or are not distributed:

- ◆ [“Attributes Distributed” on page 528](#)
- ◆ [“Attributes Not Distributed” on page 529](#)
- ◆ [“Attributes Sent Only Once” on page 529](#)
- ◆ [“Attributes Modified” on page 529](#)

Attributes Distributed

If they can be modified in ConsoleOne, all attributes not listed in the following three sections are distributed as they exist in the golden Application object. These attributes are read from the golden

Application object when building the Distribution and are sent every time ZfS creates or updates the distributed Application object.

All attributes contained in a golden Application object, not just the updated attributes, are updated in the distributed Application objects when a Distribution is rebuilt, sent, and extracted. This means that all distributed Application objects are kept in sync with their golden applications, except as noted in the next three sections.

Attributes Not Distributed

The following attributes (listed by eDirectory attribute name) are never read by the Distribution building process, and are not populated by ZfS in the distributed Application object:

Attribute Name	Location in the Application Object's Properties
App:FS Rights Path	Common tab > File Rights subtab > Path column.
App:FS Rights Volume	Common tab > File Rights subtab > Volume column.
App:Printer Ports	Common tab > Drives/Ports subtab > Ports to be Captured list box.

This list includes only those attributes that you can modify in ConsoleOne.

Attributes Sent Only Once

The following attribute (listed by eDirectory attribute name) is sent only once to provide an initial contact list:

Attribute Name	Location in the Application Object's Properties
App:Contacts	Identification tab > Contacts subtab.

This attribute is not updated by any subsequent Distribution updates. This prevents changes to this attribute in the distributed Application object from being overwritten by an original or updated contacts list in the golden Application object.

This attribute can be modified in ConsoleOne.

Attributes Modified

The following attributes (listed by eDirectory attribute name) are read from the golden Application object when the Distribution is built, but are modified to fit the Application object's new environment when the distributed Application object is created in the target server's working context:

Attribute Name	Location in the Application Object's Properties
ACL	NDS Rights tab > Trustees of This Object subtab.
App:Alt Back Link	Fault Tolerance tab > Remote Alternate App subtab.
App:Associations	Associations tab.
App:Back Link	Run Options tab > Application Dependencies subtab > Show Chain button.
App:Fault Tolerance	Fault Tolerance tab > Fault Tolerance subtab.

Attribute Name	Location in the Application Object's Properties
App:Load Balancing	Fault Tolerance tab > Load Balancing subtab.
App:Site List	Distribution tab > Link Up Site List button (which only displays if the ZfS snap-ins are installed in ConsoleOne). For how to use this button, see Step 13 on page 544 .
Application GUID	Distribution Options tab > Options subtab > GUID field.
creatorsName	Listed on the Other tab (you must click Show Read Only to view).
modifiersName	Other tab (you must click Show Read Only to view).
Object Class	Listed on the Other tab.
Revision	Listed on the Other tab (you must click Show Read Only to view).
Used By	Listed on the Other tab (you must click Show Read Only to view).

This list includes only those attributes that you can modify in ConsoleOne, and they are only displayed in an Application object when needed to define the application.

Continue with [“Maintaining Associations When Distributing Objects” on page 530](#).

Maintaining Associations When Distributing Objects

When configuring a Desktop Application Distribution, you can specify to maintain associations. This means that you want attribute associations set in the golden Application object to be maintained in the distributed Application object that is created by the Distribution.

The Desktop Application Distribution requires some manual processes, such as adding the applicable users or workstations to the distributed Application object, which is empty of this information in Desktop Application Distribution object. This is because users and workstations can be different for each server receiving a distributed application.

If you select the Maintain Associations option, then attribute associations are handled in the following ways:

- ◆ **Maintained:** User Group, Workstation Group, Organization, and Organizational Unit objects.

These are trusted groups and containers (within the source root container). They are maintained in the following manner:

- ◆ **Created New:** Group and container objects, if they do not exist.

You need to manually populate them with the users and workstations who need the distributed applications.

- ◆ **Not Overwritten:** Group and container objects, if they already exist.

If group and container objects already exist and have been assigned to the distributed Application object, those settings are not overwritten, because they could already be populated with the users and workstations that need to use the distributed applications. If you want to add other users or workstations to existing groups or containers, you must add them manually.

- ◆ **Not Created:** User and Workstation objects.

You can add the applicable users and workstations to the distributed Application objects after the Distribution has been extracted.

The Maintain Associations option is required when you distribute chained application information and folders. This is explained under [“Chained Applications in Distributions” on page 538](#).

Continue with [“Maintaining Application File Rights” on page 531](#).

Maintaining Application File Rights

File rights that you set in a golden Application object are not passed to the distributed Application objects, because file locations vary from server to server and cannot be anticipated.

The Desktop Application Distribution requires some manual processes, such as adding additional rights for file access. These processes are in addition to the minimums set by ZENworks when creating a distributed Application object. (The minimum rights might be enough for most applications.)

Review the following sections to understand how file rights are handled in Desktop Application Distributions:

- ◆ [“File Rights Are Not Distributed” on page 531](#)
- ◆ [“File Rights and Groups” on page 531](#)
- ◆ [“Chained Applications and File Rights” on page 531](#)
- ◆ [“Setting File Rights” on page 532](#)
- ◆ [“Setting Trustees and Shares Instead of File Rights” on page 532](#)

File Rights Are Not Distributed

File rights that are explicitly assigned in the Application object using the Rights to Files and Folders tab are not transferred, but are reset to the minimum necessary (Read and File Scan) for users to use the distributed applications. They are set when the distributed Application object is both created and then associated to a container or group.

File rights assigned in the Common > File Rights tab in the Application object are also not distributed.

You can later grant additional rights on these tabs in the distributed Application object and ZENworks does not remove or replace them.

File Rights and Groups

If a user or workstation is a member of a group that is distributed in the Desktop Application Distribution, then individual file rights for the user or workstation do not need to be set. The user or workstation obtains its rights to the application by virtue of its membership in the group.

Chained Applications and File Rights

If a chained application is used, all applications in the chain that require rights to a directory must be associated to a user or workstation group or a container in the golden Application’s tree structure, because individual user or workstation objects’ rights are not maintained in distributed Application objects.

Setting File Rights

To provide the Read and Write access rights to the files belonging to the chained application, in the Rights to Files and Folders tab in the User or Workstation Group object, assign the file rights.

Setting Trustees and Shares Instead of File Rights

ZfS does not set individual rights on files for NetWare®—only trustees are set on the directories that contain the files, and rights are always Read and File Scan. Therefore, on NetWare servers you should grant users Read rights to the directory where the application's files are distributed. For example, if you have the files copied to the \apps directory, users would need Read rights to the \apps directory in order to use the application whose files were copied there.

ZfS also does not set file rights in Windows. Therefore, you should set up individual shares for users to have access to the application's distributed files.

Continue with [“Subscriber Working Context Conflicts” on page 532](#).

Subscriber Working Context Conflicts

Whether your Subscribers all use the same working context or a unique working context depends on your application distribution design needs. You might have all of the Subscribers who receive a Desktop Application Distribution use the same working context if you want load balancing or fault tolerance to be used. For more information, see [“The Purpose of Desktop Application Distributions” on page 525](#).

Where there are multiple Subscribers using the same working context, an eDirectory collision is possible. In other words, multiple Subscribers cannot extract their copies of the same Desktop Application Distribution at the same time to the same working context in the tree.

For example, if two Subscribers extract an application at the same moment and create an Application object in two different eDirectory replicas, this causes a problem in eDirectory synchronization. When eDirectory finds the two different objects, but with the same name and the same timestamp in the two different replicas, eDirectory resolves this by renaming one of the objects by appending a number to the collision object's name (for details, see [Technical Information Document 10062001 \(http://support.novell.com/cgi-bin/search/searchtid.cgi?/10062001.htm\)](http://support.novell.com/cgi-bin/search/searchtid.cgi?/10062001.htm)).

If you use the same working context for a group of Subscribers, then you must make sure that each Subscriber's Extract schedule fires at a different time, allowing enough time between these schedules for extraction to be completed by a Subscriber before the next Subscriber begins extracting.

If you have each Subscriber use a unique working context, all Subscribers can then extract their copies of the same Desktop Application Distribution at the same time, and no eDirectory collisions occur.

If a Distribution is set to extract immediately, the same scenario can exist.

Continue with [“Maintaining Source Paths” on page 532](#).

Maintaining Source Paths

Many applications require supporting files, and the paths to those source files must be established in the Application objects in ZfD. This is known as the “source path.”

This section applies to Desktop Application Distributions containing Application objects that use source paths. For applications that require only an executable file (such as notepad.exe), source paths are not required in their Application objects.

Review the following sections to understand how source paths are used in Desktop Application Distributions:

- ◆ [“Source Path Usage in ZfS” on page 533](#)
- ◆ [“Purpose of the SOURCE_PATH Macro” on page 535](#)
- ◆ [“How the SOURCE_PATH Macro’s Values Are Interpreted” on page 535](#)
- ◆ [“Defining a Variable in ZfS” on page 536](#)
- ◆ [“Using the Source Path Option in the Distribution” on page 537](#)

Source Path Usage in ZfS

To show what happens with the attribute between the golden and distributed Application objects during the Desktop Application Distribution process, the following table lists where you can find and configure source paths in ConsoleOne, their purposes, how these locations are populated, and their distribution status.

Source Path Name (Type)	Location in the Application Object	Purpose	How Populated	Distribution Information
SOURCE_PATH (macro ¹)	Distribution Options > Application Files > Source column	Provides path resolution from the SOURCE_PATH macro.	From the SOURCE_PATH macro.	This is distributed for each application file listed under the Name column that uses it.
SOURCE_PATH (macro)	Common > Macros > Name column	Defines a source path (in the Value column) to be used by the Application object.	From entries that you make on this page when the distributed Application object is created.	This is distributed with modifications to fit the Subscriber's environment. If it is changed in the golden Application object, it is updated in the distributed Application object with the necessary modifications. This source path on the golden Application object should be kept stable, in order to avoid Novell Application Launcher distribution problems.

Source Path Name (Type)	Location in the Application Object	Purpose	How Populated	Distribution Information
Package Source List (box)	Common > Sources	Provides a list of source paths for the Load Balancing and Fault Tolerance properties to use.	From the SOURCE_PATH macro, from each Subscriber using the same working context that receives and extracts the Distribution, and from any entries you make using the Add button.	<p>Only the SOURCE_PATH macro's entry is duplicated by ZFS using the long (DNS) version of the path.</p> <p>The listed source paths must be either valid UNC paths or variables that resolve to valid UNC paths.</p> <p>Each listed source path points to a complete set of the application's files that are located on the Distributor server's file system. (The Distributor cannot gather its Desktop Application Distribution's files from other servers.) These actual source files pointed to by the source paths are overwritten every time the Distribution is rebuilt and sent again.</p> <p>This field on the distributed Application object is cumulative, and is not overwritten when the Distribution is re-sent. Its entries come from selecting Load Balancing or Fault Tolerance for the Subscribers receiving the Distribution that use the same working context, or your use of the Add button on the distributed Application object. However, if you make a change to the SOURCE_PATH macro in the golden Application object, that source path is updated in this list box and is inserted first in the list. The previous source path is not replaced, but is left in the list. It is no longer valid, and you can delete it.</p>
Source List (box)	Fault Tolerance > Fault Tolerance	<p>Provides a list of servers that can provide redundancy in case a server being used for Novell Application Launcher work goes down.</p> <p>All source paths listed must point to identical application file sets; otherwise, the distributed applications can fail to be created correctly.</p>	From each Subscriber using the same working context that receives and extracts the Distribution.	This information is not distributed from the golden Application object. You must populate this field by sending the Distribution to multiple Subscribers that are using the same working context.

Source Path Name (Type)	Location in the Application Object	Purpose	How Populated	Distribution Information
Source List (box)	Fault Tolerance > Load Balancing	Provides a list of servers that can provide load balancing among them for doing Novell Application Launcher work. All source paths listed must point to identical application file sets; otherwise, the distributed applications can fail to be created correctly.	From each Subscriber using the same working context that receives and extracts the Distribution.	This information is not distributed from the golden Application object. You must populate this field by sending the Distribution to multiple Subscribers that are using the same working context.

¹ A “macro” in ZfD has the same functionality as a “variable” in ZfS.

Purpose of the SOURCE_PATH Macro

The SOURCE_PATH macro defines the source path in an Application object to where its application’s files reside on the Distributor server’s file system. This is where the Distributor accesses the application files for building the Desktop Application Distribution.

The SOURCE_PATH macro’s value is used to create the location on the Subscriber server’s file system where those application files are to be placed by the Subscriber when it creates the distributed Application object.

The information in the value of the SOURCE_PATH macro includes:

- ◆ Server identification (the Distributor server) in either the server name, IP address, or full DNS name
- ◆ Volume or drive on the Distributor server
- ◆ User-defined path information (if provided in the wizard)
- ◆ Application path information (selected in the wizard)

Some examples:

```
server1.novell.com\sys\apps\acrobat
```

```
server1.novell.com\n\apps\acrobat
```

This is resolved to a valid UNC path, such as:

```
\\server1\sys\apps\acrobat
```

If you include a macro (or variable) within the value of the SOURCE_PATH macro, ZfS does not resolve that embedded information. ZfS only resolves the SOURCE_PATH macro’s value to a valid UNC path.

How the SOURCE_PATH Macro’s Values Are Interpreted

Tiered Electronic Distribution searches variables to find a match with the golden Application object’s source path. For example, if the source path in the golden Application object is n:\apps\acrobat, the following order is searched to find a match:

```
n:\
N:\
```

n:
N:
n
N

However, if the golden Application object's source path is N:\apps\acrobat, the following order is searched to find a match:

N:\
n:\
N:
n:
N
n

Defining a Variable in ZfS

Historically, mapped drives have been embedded into an Application object as a means of launching that application from a mapped drive on the desktop. ZfS uses variables to distribute applications that use drive mappings.

Because volume names and mapped drives for the Distributor and all of the Subscribers receiving the Distribution can be different, variables allow you to identify these locations with a value that is interpreted by the Distributor and each Subscriber.

You can define variables globally and individually:

- ◆ **Globally using the Tiered Electronic Distribution policy in the Service Location Package**

Variables defined in the Tiered Electronic Distribution policy (Service Location Package) are available to all Subscriber objects associated with the policy, such as associating the policy package to the parent containers of the Subscriber objects. For the policy to be in effect for each Subscriber, make sure on the Variables property page that the Include Policy check box is selected.

The policy package must also be associated with the parent container of the Distributor object. The variable definition in the policy ensures that the Distributor knows where to gather the application files from.

If both the Distributor and Subscribers use the same variable value, then only one Tiered Electronic Distribution policy is needed, and you can associate its Service Location Package to the parent containers of both the Distributor and the Subscribers.

For example, the mapped drive source path for a golden Application object is n:\applications\acrobat and you want n: to represent the sys:\public directory on the Distributor server. To create the variable, in the Tiered Electronic Distribution policy select the Variables tab, then enter the n: for the variable and sys:\public for its value. Then, the Distributor can find the \applications\acrobat directory on its sys: volume when it needs to build the Distribution.

For more information on this policy, see [“Tiered Electronic Distribution” on page 471](#).

- ◆ **Individually in each target Subscriber's properties**

You can define a variable for any Subscriber object. This definition overrides the same variable if it is defined in a Tiered Electronic Distribution policy that the Subscriber is associated with.

This is useful for when the Subscriber server's volume name or mapped drive is different than the Distributor server's (so they can't use the same Tiered Electronic Distribution policy), or you have a variety in volume names or mapped drives among the Subscribers receiving the Distribution.

For information on how to define variables on Subscribers, see [“Creating a Variable” on page 582](#).

Using the Source Path Option in the Distribution

If a golden Application object uses mapped drives, enable the Keep the Same Source Paths for the Replicated Objects option when running the Desktop Application Distribution Wizard. Enabling this option causes the Distribution to retain a golden application's source paths for when a mapped drive designation must be used by the application that is distributed. The value of the mapped drive determines where the applications' files are copied.

If the golden applications' source paths are mapped drives, but you want the distributed applications to use a UNC path according to the extraction directory, then you do not need to select this option, but you must define the Tiered Electronic Distribution policy in the Service Location Package with variables defined for the mapped drives. This package must be associated to the Distributor with the variable defined in order for the Distribution build to work. It should also be associated with containers for the Subscriber objects, or any container above them.

Key points about this option:

- ◆ If a golden Application object's Package Source List box contains a mapped drive, you must enable the Keep the Same Source Paths for the Replicated Objects option. The mapped drive letter is treated like a variable that needs to be resolved on both the Distributor and Subscriber to complete the valid UNC path.
- ◆ If a golden Application object's Package Source List box contains a drive mapping that is local to a server other than the Distributor server, no application files can be gathered or distributed, because all files to be included in the Distribution must be contained on the Distributor server's file system.
- ◆ Enabling this option affects all Application objects in the Distribution, including chained applications. Therefore, all mapped drive properties for each of the Application objects included in the Distribution are distributed to keep their golden Applications' source paths, and each application and chained application must have mapped drives for the source path.
- ◆ If you select this option, only the Application object's Default Directory Path is used, because the Application Destination Directory Path field in the next wizard page is disabled. Therefore, you cannot change the path.
- ◆ If you select this option, or leave it unselected, that choice becomes the permanent use of this option for the Distribution. This is done to prevent problems that can occur from alternating between using and not using an Application object's mapped drives.
- ◆ For chained applications, source paths are treated the same for all chained applications as they are for the first application that the others are chained to.

Miscellaneous Issues

- ◆ [“Application Dependencies and Requirements” on page 538](#)
- ◆ [“Chained Applications in Distributions” on page 538](#)
- ◆ [“Site Distribution Objects” on page 538](#)

Application Dependencies and Requirements

Dependencies and requirements can be confusing with regard to the distribution of attributes:

- ◆ **Dependency:** An application dependency, such as a chained application, is updated in a distributed Application object when the Distribution is rebuilt, sent, and extracted.

To view application dependencies: in the properties of an Application object, click Run Options > Application Dependencies.

- ◆ **System Requirement:** An system requirement, such as an operating system for the application to run on, is updated in a distributed Application object when the Distribution is rebuilt, sent, and extracted.

To view system requirements: in the properties of an Application object, click Availability > System Requirements.

One exception is that an application requirement is not updated in a distributed Application object when the Distribution is rebuilt, sent, and extracted. Instead, we recommend using application dependencies.

Chained Applications in Distributions

If multiple applications contain the same chained application, the application's files are only contained once in the Distribution. This reduces the Distribution's file size.

For example, if you distribute several icons (each its own Application object) that each require an office software suite, that suite software is only included once in the Distribution.

If your Desktop Application Distribution has chained applications, you must enable the Maintain Associations option when configuring the Distribution.

Chained applications in Distributions are only available in ZENworks for Desktops 4.x and later.

Site Distribution Objects

ZfS 3.0.2 does not use a Site Distribution object. Previous versions of ZENworks might have used Site Distribution objects with this Distribution type.

Requirements

The following requirements must be met before creating and sending Desktop Application Distributions using Tiered Electronic Distribution:

- Desktop Application Distributions can be sent to only NetWare and Windows servers. This Distribution type is not supported on Linux and Solaris servers.
- In order to use Novell Application Management with ZENworks for Servers 3.0.2, you must have ZENworks for Desktops 3.x or 4.01 installed. However, chained applications in Desktop Application Distributions are only supported in ZENworks for Desktops 4.01 and later.
- For Windows NT/2000 servers that have eDirectory installed on them, to be able to send a Desktop Application Distribution to these servers you must install Policy and Distribution Services by browsing for the server object in the eDirectory listing, not by browsing for the computer object in the Microsoft* domains listing.
- ZfD and ZfS must both be installed to the same tree, including their respective schema extensions.

- ❑ For golden Application objects to be functional in a Desktop Application Distribution, the snap-ins for both ZfS and ZfD must be installed in ConsoleOne.
- ❑ Make sure all of the associations in the golden Application object are in the source root context or below.

IMPORTANT: If even one of your associations is outside the source root context, the Distributor fails to build the Distribution.
- ❑ For Windows servers, you must have a shared location established for extracting the Distribution's files, where all users can have access to those files.
- ❑ The source path must point to application files that are located on the Distributor server's file system, because the Distributor cannot gather files from other servers' file systems.

If the Package Source List box contains a local drive mapping, no application files are gathered or distributed.

If the Package Source List box contains a mapped drive, Keep the Same Source Paths for Replicated Objects must be selected. The drive letter is treated like a variable that needs to be resolved on both the Distributor and Subscriber to complete a valid UNC path.

Use a policy to define the variables on a Distributor. On the Subscriber you can use either the variable list in the Subscriber object, or a policy that is associated to the container where the Subscriber object resides.
- ❑ The Subscriber object must have the Working Context attribute defined. This is the eDirectory context where the Subscriber creates the objects related to the Desktop Application Distributions that it receives.

Multiple Subscribers can use the same working context if you intend to use them for load balancing or fault tolerance.
- ❑ Under some circumstances, you might receive a Java Out of Memory error when a Desktop Application Distribution is extracted. To prevent this, edit the ted.ncf file on the Subscriber server and change the memory variable on the last line from 128 to 256. Then the Distribution should extract. However, this treats the symptom and not the cause, which you might eventually need to resolve.

Creating a Desktop Application Distribution

- ◆ [“Understanding the Desktop Application Distribution Wizard” on page 539](#)
- ◆ [“Creating the Distribution” on page 540](#)

Understanding the Desktop Application Distribution Wizard

ZENworks uses Tiered Electronic Distribution to distribute Application objects to other locations in the same tree or other trees. Using a Desktop Application Distribution, the original files associated with the applications are copied to the appropriate server locations where they can be used to locally service user groups and workstation groups associated with the distributed Application objects.

To distribute applications created in ZfD, you use the ZfS Desktop Application Distribution Wizard to configure the Distribution. This includes:

- ◆ Determining the destination's tree context

- ◆ Determining whether to maintain the associations between user/workstation groups or containers and the applications
- ◆ Determining whether to have automated load balancing or fault tolerance
- ◆ Determining how to trigger rebuilds of the Distribution
- ◆ Selecting the applications
- ◆ Determining the file copying paths

To create a Desktop Application Distribution using the wizard, continue with [“Creating the Distribution” on page 540](#).

Creating the Distribution

- 1** Fulfill all of the requirements listed under [“Requirements” on page 538](#).
- 2** In ConsoleOne, right-click the container where you want the Distribution object located > click New > click Object > select TED Distribution > click OK.
- 3** Enter a name for the Distribution.

IMPORTANT: Periods (.) are not allowed in Distribution names. Instead, use dashes (-) or underscores (_) as word separators. If you use a period in the Distribution name, the Distribution will not be sent, and the Distributor will not reload after it has been exited.
- 4** To give a Distributor ownership of the Distribution, browse and select the Distributor object > click Define Additional Properties > click OK.

The Distribution object’s properties are displayed.

- 5** Click the General tab > fill in the Settings fields:

Active: Required. In order to make a Distribution available to Subscribers, it needs to be active.

Use Digests: Digests are used by Distributors and Subscribers to verify that Distributions have not been tampered with while in transit. The digest provides an ND5 checksum for the Subscriber to compare.

Digests also detect corruption in a Distribution’s package. If corruption is present, the Subscriber renames the distfile.ted Distribution file to distfile.corrupt and the Distribution is rebuilt and sent the next time the Channel’s schedule fires.

Encrypt: You can have the Distribution encrypted if you will be sending it across non-secured connections. Encryption provides security for the Distribution during transit between the Distributor and Subscriber when they are not within the same firewall. Click either Strong or Weak encryption.

You also must have the same version of NICI 2.4 installed to each of these servers for encryption to work (see [“Installing NICI 2.4” on page 341](#)).

Maximum Revisions: This number helps you to control disk space usage by determining how many versions of a particular Distribution are kept in the Distributors’ and Subscribers’ working directories. The default is 10. Increase the number if data is changing often and the changes are minimal (smaller delta files). Decrease the number if data is not changing very often, or if a significant amount of data is changing (larger delta files). If you select 1, the Delete Previous Revision field will be checked.

Delete Previous Revision Before Receiving Next: This option is available if you selected 1 as the number for the Maximum Revisions field. If the Distribution is so large that it might

compromise the available disk space on the Subscriber server, you can conserve disk space by checking this option, which causes the previous version to be deleted before receiving the next version. If you leave the check box empty, the new version is received in its entirety before the older version is deleted. Either way, you have only the one version of the Distribution in the Subscriber's working directory after the Distribution has been received.

Priority: You can give the Distribution a priority that determines how it is sent in relation to other Distributions. A High priority means it is sent before Medium or Low priority Distributions.

Distributor: Displays the DN of the Distributor object that builds and sends this Distribution. You selected the Distributor when you created the Distribution object.

Description: Provide useful details about the Distribution, such as the name of the desktop application, the files and directories it contains, intended user groups, and so on.

6 Click the General tab > click Restrictions.

You can select whether to have platform restrictions for the Distribution itself. This is not a restriction for the distributed Application object.

No Restrictions: This option is checked by default. To determine platform restrictions, click this radio button to disable it > click the check boxes corresponding to the platforms you want to receive this Distribution.

Platforms with check boxes are not checked cannot receive the Distribution. In other words, you restrict sending to a platform by disabling the No Restrictions option and not selecting the platform.

The available options are:

No Restrictions	NetWare 5.1	Solaris
NetWare All	NetWare 5.x	Linux
NetWare 4.x (ZfS 2)	NetWare 6.x	
NetWare 5.0 (ZfS 2)	Windows Server	

No Restrictions means that the Distribution can be sent to any platform.

If you select NetWare All, you do not need to select any of the individual NetWare platforms.

7 Click the Type tab > in the Select Type drop-down box, select Desktop Application > click Setup.

The Desktop Application Distribution Wizard is started.

You can also start this wizard from the Desktop Application Agent properties page by clicking Modify.

7a Click Next after reading the Introduction information.

7b Fill in the fields > click Next.

Maintain Source Tree Structure: Duplicates the source tree's structure at the destination's location (the target Subscriber's working context) for placing the distributed application objects. If you will be selecting chained applications, you must check this option.

For more information, see [“Maintaining Source Paths” on page 532](#).

Maintain Associations: Distributes the associated groups or containers at the target location if they do not exist. However, users or workstations contained in the groups or containers in the source location are not distributed.

For more information, see [“Maintaining Associations When Distributing Objects” on page 530](#).

You must enable this option if you have chained applications in the Distribution. For more information, see [“Chained Applications in Distributions” on page 538](#).

IMPORTANT: Rights previously set in the associated user/workstation groups or containers that are maintained are not distributed, but set to the minimum necessary in the distributed groups or containers so that users can use the applications.

Source Root Container: Select a container to be used as the root container for the golden application objects to be distributed. You should only select golden Application objects from this root container and its subordinate containers.

Load Balance and Fault Tolerance Support: Choose whether to use automated load balancing, fault tolerance, or neither:

- ◆ **Load Balance:** Automates spreading server workloads over the servers being used for the Desktop Application Distributions. The functionality of fault tolerance (redundancy) is automatically accomplished through load balancing.
- ◆ **Fault Tolerance:** Allows a server being used for Desktop Application Distributions to assume the distribution duties of another server that goes down. Fault Tolerance does not provide load balancing.
- ◆ **None:** Neither option is applied. You must individually configure each distributed Application object for load balancing or fault tolerance, if you want that support on an individual basis.

For these two features to work:

- ◆ Multiple Subscribers receiving the Distribution must be using the same working context
- ◆ The User Source List button must be selected on the Fault Tolerance > Fault Tolerance or the Fault Tolerance > Load Balancing properties pages of the Application object

Depending on the selected options, the Load Balancing or Fault Tolerance pages are populated with the file locations on all servers that share this working context.

For more information, see [“Distributed Applications in ZfS” on page 526](#).

7c Click Add to browse for and select golden Application objects > click Next.

Do not browse above the root directory that you established in the previous wizard page, especially if you have checked the Maintain Source Tree Structure option.

IMPORTANT: The Desktop Application source files must reside in the Distributor server's file system. The Distribution cannot be gathered from another server's file system.

7d Enter the destination volume or shared folder.

The application files distributed are those that are associated with the golden Application objects you selected in the previous wizard page.

You can enter a variable instead. If you use a variable, it must be defined in the destination Subscriber server's properties to point to the target server's volume or shared folder.

This volume (NetWare) or shared folder (Windows) becomes the root location for placing subordinate directories where the application files will be copied.

- 7e** To use only an application's default path, click Default Application Directory Path, which is placed beginning with the root location you specified in [Step 7d](#).

or

To enter a user-defined directory path to the application's files, click User-Defined Directory Path > enter your path information.

The path you specify is used in the following manner:

- ◆ The volume or shared folder name remains unchanged (as specified in [Step 7d](#)).
- ◆ Your path information is inserted after the volume or shared folder name.
- ◆ Part of the application's default path is appended to your path information, beginning with the default path's immediate parent directory to the application's files. Any default path information that was above the immediate parent directory is replaced by your path entry.

The result is a customized directory path that begins with the volume or shared folder, has your user-defined path information next, and ends with the application's immediate directory. For example, the default path to the application's executable file (APPLICATION.EXE) might be:

```
\Application_Root_Directory\Application_Subdirectory
```

and you enter MyPath for your user-defined path, the new full path to the executable will now be:

```
C:\MyPath\Application_Subdirectory\APPLICATION.EXE
```

where you entered C: as the shared folder, MyPath as your user-defined path, Application_Root_Directory is replaced by MyPath, and Application_Subdirectory is the immediate parent directory to the executable, APPLICATION.EXE.

- 7f** Click Next to continue.

The Summary page is displayed.

- 7g** To make changes, click Back.

- 7h** When you have finished configuring the Distribution object, click Finish to exit the wizard.

You can edit the Distribution at any time on the Type tab of the Distribution object by clicking Modify.

- 8** Click the Channels tab > click Add > browse for and select the Channel for this Distribution.

Each Distribution must be associated with at least one Channel if it is going to be used to push data to a Subscriber. A Distribution is sent to all Subscribers that are subscribed to the selected Channel.

- 9** Click the Schedule tab > select a Build schedule:

“Never” on page [572](#)

“Daily” on page [571](#)

“Monthly” on page [571](#)

“Yearly” on page [573](#)

“Interval” on page [571](#)

“Time” on page 573

“Run Immediately” on page 572

- 10** Click Apply to create the Distribution.

You are prompted to copy additional security certificates.

- 11** Click Yes to resolve the certificates.

This copies the security certificates from the Distributor to Subscriber subscribed to the Channel.

For information, see “Resolving Certificates” on page 552.

- 12** Click OK to close the Distribution object.

The next time the Distributor reads eDirectory (this schedule is set in the Distributor object’s properties), it retrieves all of the information about the new Desktop Application Distribution, such as Distribution details, the Build schedule, and so on.

The Distribution is built according to the Build schedule, sent according to the schedule set in the Channel object, and extracted according to schedule set in the Subscriber object.

If the Distributor throws an exception during the file gathering process, the Distribution is not built. The Distributor logs the failure in the reporting database.

If the Subscriber throws an exception during extraction, the process is not completed. The Distributor receives this information from the Subscriber and logs the failure in the reporting database.

After extraction, ZfD users whose objects are located in the associated containers, or are members of a distributed group, will have access to the desktop applications that were distributed.

IMPORTANT: For Desktop Application Distributions, a built-in delay exists to accommodate directory synchronization when you have multiple applications being distributed at the same time (whether by one or multiple Distributions).

Subscribers can receive Desktop Application Distributions all at the same time, but extract them one at a time. And, when there are multiple applications contained in one Distribution, the Subscriber creates the distributed Application object and copies the files one application at a time. The built-in delay helps directory synchronization for the newly-created Application objects to occur smoothly.

To determine how much additional time this built-in delay might add to the distribution process, multiply each application contained in a Desktop Application Distribution by 30 seconds.

As a rule of thumb, if an application being distributed includes multiple versions, such as one baseline and two deltas, each of these three versions receives the same 30-second delay. For example, if you are sending 10 desktop applications, and each has three versions, the completion of the Distribution extractions would take at least 15 minutes.

- 13** So that Desktop Management users and workstations can have automatic access to their applications from any geographic location, you must link up the site lists:

- 13a** Wait for the Desktop Application Distribution to be distributed and extracted by each Subscriber server that received it, because the distributed Application objects must be created and the application’s files installed before you can link up the site lists.

- 13b** In ConsoleOne, right-click the golden Application object that was used to build the Distribution > click Properties.

- 13c** Click the Distributions tab > click the Link Up Site Lists button.

All distributed Application objects that were created from the golden Application object are displayed in the Replicated Applications list box, and all Distributions containing the distributed Application objects are listed in the Distributions Currently In list box.

What the Link Up Site List button does:

1. For the golden Application object, it searches for each distributed Application object that was created from the Distribution and lists the full DN of those objects in the golden Application object's properties (in the Replicated Applications list box).
2. For each distributed Application object, it searches for the other distributed Application objects that were created from the Distribution, and the golden Application object, and lists the full DN of all of these objects in the distributed Application object's properties (in the Replicated Applications list box).
3. This process is repeated for each distributed Application object.
4. For each Application object listed in the Replicated Applications list box, any Distributions associated with those objects are listed in the Distributions Currently In list box in each of these Application objects.

Thus, the golden Application object and all distributed Application objects have each other listed in their Replicated Applications list box, which allows users to have local access to the same application no matter where they connect to their network.

13d Click OK to close the golden Application object's properties.

13e Repeat [Step 13b](#) through [Step 13d](#) for each golden Application object that was used to build the Desktop Application Distribution.

You only need to perform the site list link-up on the golden Application objects.

Rebuilding Desktop Application Distributions

The following sections explain the different issues with rebuilding Desktop Application Distributions, including how to trigger a rebuild:

- ♦ [“All Attributes Are Updated” on page 545](#)
- ♦ [“Triggering a Rebuild” on page 545](#)

All Attributes Are Updated

All attributes contained in a golden Application object, not just the modified attributes, are updated in the distributed Application objects when a Distribution is rebuilt, sent, and extracted. This means that if you make a change to an attribute in a distributed Application object, such as a source path, that source path is overwritten by the source path data in the golden Application object. In other words, all distributed Application objects are kept in sync with their golden Application object. Exceptions to this are described in [“Maintaining a Golden Application's Attributes” on page 528](#).

A rebuilt Desktop Application Distribution includes all file changes made after the last time the Distribution was built.

Triggering a Rebuild

You can control when a Distribution is rebuilt in the following ways:

- ♦ [“Modifying an Object” on page 546](#)

- ◆ “Removing a Distributed Application Object” on page 546

Modifying an Object

When you modify a Distribution object or one of its golden Application objects, its internal revision number is automatically changed, which triggers a rebuild of the Distribution according to its established Build schedule.

Modifications include adding or removing applications from the Distribution. However, if you simply update, add, or remove application files in the Distributor server’s file system, this does not alter the internal revision number of the Desktop Application Distribution object. ZENworks’ file synchronization feature does not apply to the files in Application objects. Therefore, no rebuild is triggered.

If you should add, remove, or update any files belonging to a golden Application object, those changes are included when the next rebuild is triggered.

Removing a Distributed Application Object

Removing a distributed Application object causes a backlink to the golden Application object to change without any other changes being made to the object. This causes the internal revision number to change on the golden Application object, which triggers a rebuild of its Distribution according to the established Build schedule.

Cleaning Up Desktop Application Distribution Files

Tiered Electronic Distribution is not designed to clean up ZfD files, for example, when you might delete a golden Application object in ZfD. The distributed Application object created by the Desktop Application Distribution is not automatically deleted. You must manually remove this eDirectory object and the related application files on the server’s file system.

To do this, search in ConsoleOne for the Application object and note its filename before deleting the golden Application object. This can make manually cleaning up easier after deleting a golden application.

You can verify that the distributed application exists after you’ve deleted the golden Application object and its files, and then remove the distributed version:

- 1** In ConsoleOne, delete the golden Application object.
- 2** In a file browser, delete the files related to the application.
- 3** In ConsoleOne, right-click the Distributor object associated with the golden Application object’s Distribution.
- 4** Click Refresh Distributor from the drop-down menu, then click Yes > OK.
- 5** Check the destination location for the distributed Application object.
It should still be present, even though it no longer exists in the Distribution.
- 6** In ConsoleOne, delete the distributed Application object.
- 7** In a file browser, delete the files related to the distributed application.

Sending Desktop Application Distributions Tree-To-Tree

Desktop Application Distributions can be sent between trees. However, you must do the following for this to work:

- 1** Create an External Subscriber object in the Distributor's tree that points to the target server in the other tree where you want to send the Desktop Application Distribution.

This will enable the Distributor server to send the Distribution directly to the target server using the IP address listed in the External Subscriber object.

- 2** The target server that is to receive the Desktop Application Distribution must have a Subscriber object in its own tree, so that it will have the rights to eDirectory for creating the new Desktop Application object in that tree.

This Subscriber must be within a working Tiered Electronic Distribution system.

- 3** Set the working context in the Subscriber object for the target server, if this was not done during installation.

If the working context is not set for the target server, authentication will fail during the extraction process.

- 4** Create the Desktop Application Distribution (see [“Creating a Desktop Application Distribution” on page 539](#)).

Defining the Desktop Application Distribution is the same process, whether it is being sent within a tree or across trees.

- 5** Add the External Subscriber object to the Channel where the Desktop Application Distribution is listed.

- 6** Manually copy the certificates to the Subscriber in the target tree.

- 7** Send the Distribution.

20 Security in Policy and Distribution Services

Novell® ZENworks® for Servers (ZfS) provides the following types of security for Policy and Distribution Services:

- ◆ [“Distribution Security Using Signed Certificates and Digests” on page 549](#)
- ◆ [“Distribution Security Using Encryption” on page 558](#)
- ◆ [“Security for Inter-Server Communication Across Non-Secured Connections” on page 562](#)

Distribution Security Using Signed Certificates and Digests

Policy and Distribution Services uses signed certificates to validate whether Distributions are from a trusted source, or have been tampered with. This security is automatically used by Policy and Distribution Services for all Distributions. However, there are actions you might need to take to get Policy and Distribution Services to create and process the certificates.

Policy and Distribution Services also provides optional Distribution security with digests. A digest is used by the Subscriber to determine whether a Distribution has been tampered with after it left the Distributor.

There are two features of TED that deal with security:

- ◆ **Certificates:** Security certificates (required) are issued by each Distributor to all Subscribers receiving its Distributions. In order for a Subscriber to accept its first Distribution from a Distributor, it must have a certificate in its security directory from that Distributor. After receiving its first Distribution from the Distributor, the certificate is then stored in the .KEYSTORE file. The content of the .KEYSTORE file can be viewed in iManager.

For information on security certificates for encrypted Distributions, see [“Distribution Security Using Encryption” on page 558](#).

- ◆ **Digests:** Digests (optional) can be created for each Distribution at the time it is built. The digest is used by the Subscriber to determine whether a Distribution has been tampered with after it left the Distributor.

The following sections provide more information on understanding, creating, and using certificates and digests:

- ◆ [“Understanding Digests” on page 550](#)
- ◆ [“Understanding Certificate Usage in Policy and Distribution Services” on page 550](#)
- ◆ [“Important Points about Certificates” on page 551](#)
- ◆ [“ConsoleOne User Rights and Certificate Copying” on page 552](#)
- ◆ [“Certificate File Locations” on page 552](#)
- ◆ [“Resolving Certificates” on page 552](#)

- ◆ [“Handling Invalid Certificates” on page 553](#)
- ◆ [“Certificate and Private Key Directories” on page 557](#)
- ◆ [“Creating Security Certificates for Non-Encrypted Distributions” on page 557](#)
- ◆ [“Manually Copying Certificates for Non-Encrypted Distributions” on page 558](#)

Understanding Digests

Important points about digests:

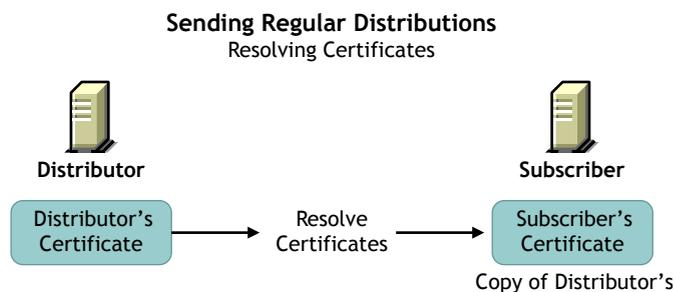
- ◆ Digests can be created for each Distribution at the time it is built. The digest is used by the Subscriber to determine whether a Distribution has been tampered with after it left the Distributor.
- ◆ The Digest option is available for all Distribution types. The Digest check box is displayed on the General tab of the Distribution object’s properties.
- ◆ A digest will add about 30% to the build time. Factors that can affect build time using digests are CPU and hard drive speeds, amount of RAM, server workload, and so on.

Understanding Certificate Usage in Policy and Distribution Services

A certificate is a security mechanism used by Policy and Distribution Services to ensure that the Distribution received by a Subscriber was actually sent by the Distributor owning that Distribution. Because configuration information can also be sent to the Subscriber, it ensures that the configuration information has been sent from a known Distributor and that the data has not changed.

All Subscribers must receive a valid security certificate from each Distributor that sends Distributions to them. Without a matching certificate, a Subscriber cannot receive Distributions from the Distributor.

The following illustrates the process of using certificates with Distributions:



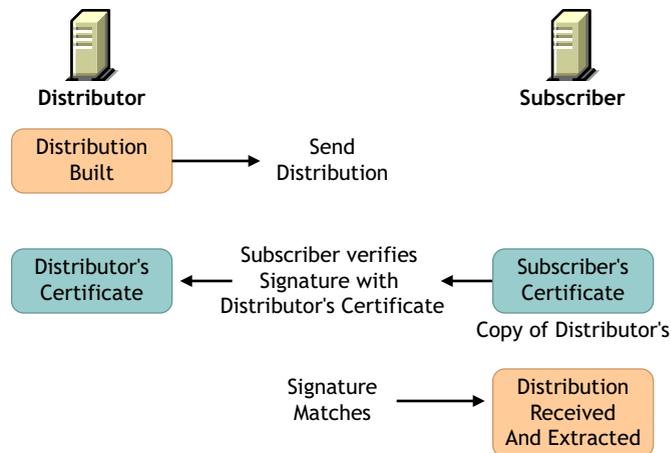
Before a Distribution should be sent, certificates must be resolved. This ensures that the Distribution received by a Subscriber was actually sent by the Distributor owning that Distribution.

For information on resolving certificates, see [“Resolving Certificates” on page 552](#).

After certificates have been resolved, the following illustrates how the Subscriber uses the certificate to ensure it is receiving a valid Distribution:

Sending Regular Distributions

Sending the Distribution



Important Points about Certificates

- ◆ Certificates are issued by each Distributor to all Subscribers receiving Distributions from that Distributor. In order for a Subscriber to accept Distributions from a Distributor, it must have received a certificate from that Distributor.
 - ◆ For security, certificate key pairs are created by the Distributor.
 - ◆ The public key is written to the Distributor server's file system, which self-signs a certificate and stores it in Novell eDirectory™.
 - ◆ The Subscriber software does not need to be running on the Subscriber server to have certificates copied to the server.
 - ◆ The association of Distributions (owned by a Distributor) and Subscribers to a Channel determine which Subscribers should receive certificates from which Distributors.
 - ◆ A Distributor will send certificates to all Subscribers that subscribe to Channels where the Distributor has Distributions.
 - ◆ A Subscriber will request certificates from all Distributors that have Distributions in Channels to which it subscribes.
 - ◆ A certificate can be passed from a Distributor to a Subscriber under the following circumstances:
 - ◆ When a Subscriber is initially subscribed to a Channel and you click OK to apply the changes.
 - ◆ When you right-click a Subscriber Object and select Resolve Certificates. The Subscriber will then request certificates from all Distributors that it will receive distributions from.
 - ◆ When a Distribution is listed in a Channel and you click OK to apply the changes.
 - ◆ When you right-click a Distributor Object and select Resolve Certificates. The Distributor will send certificates to all Subscribers that it sends distributions to.
- For information on resolving certificates, see [“Resolving Certificates” on page 552](#).
- ◆ When you add a Distribution and/or a Subscriber to a Channel. When clicking okay, the Resolve Certificates? dialog will be displayed. If you answer Yes, certificates will be sent

by all Distributors who have Distributions associated with that Channel to all Subscribers subscribed to that channel.

- ◆ Manually copying a certificate file to a transfer medium (such as a diskette or local drive), then to the ZENWORKS\PDS\TED\SECURITY directory on a server.

Basically, any time the relationship changes between the Subscribers, Channels, or Distributions, a certificate can be passed.

- ◆ If a Distributor object is deleted and re-created to point to the same server, all certificates on the subordinate Subscribers become invalid. Certificates must be deleted from the Subscriber's security subdirectory. Then the Distributor must send the new certificates to those Subscribers.
- ◆ ConsoleOne copies the certificate files to Subscriber servers. Therefore, the client software on the workstation running ConsoleOne must have access to the Subscriber servers' file systems. For Windows Subscriber servers, the Domain and Workgroup rights on the workstation must be set up to facilitate automatic certificate copying. Otherwise, a 1204a error will be given.

ConsoleOne User Rights and Certificate Copying

The administrator using ConsoleOne[®] must have sufficient rights to the Subscriber server in order for a certificate to be copied to that server when the administrator resolves certificates in ConsoleOne. This is because when you use ConsoleOne to configure a Subscriber object to receive the Distributions from a particular Channel, the Distributors owning the Distributions in that Channel must send certificates to the Subscriber's server.

For NetWare[®] Subscribers, the ConsoleOne user automatically has sufficient rights by virtue of being able to configure the Subscriber object.

For Windows Subscribers, administrator rights for the ConsoleOne user must be set up in Windows:

- ◆ **Windows NT Subscriber Server:** Select User Manager for Domains.
- ◆ **Windows 2000 Subscriber Server:** Select Active Directory Users and Computers, or select Local Users and Groups.

Certificate File Locations

Certificates are stored in the ZENWORKS\PDS\TED\SECURITY directory on each Subscriber's server.

WARNING: Make sure the ZENWORKS\PDS\TED\SECURITY directory is a non-public directory. This directory should not be read by anyone other than an administrator. The .KEYSTORE file is in the ZENWORKS\PDS\TED\SECURITY\PRIVATE directory and is by default hidden from non-administrative users.

Certificates are usually named after the fully qualified DNS name of the Distributor server, such as `Distributor_Server001.novell.com.cer` or `Distributor_Server001.novell.com.csr`. The TCP/IP address of the server would be used for .CSR files if a DNS name could not be resolved. The certificate would then be named using its IP address, such as `155.55.155.55.csr`.

Resolving Certificates

IMPORTANT: ConsoleOne copies the certificate files to Subscriber servers. Therefore, the client software on the workstation running ConsoleOne must have access to the Subscriber servers' file systems. For Windows

Subscriber servers, the Domain and Workgroup rights on the workstation must be set up to facilitate automatic certificate copying. Otherwise, a 1204a error will be given.

When you are automatically presented with the option in ConsoleOne to resolve certificates, determine the following to know whether to click Yes or No:

- ◆ If the Distributor currently has Distributions associated with this Channel, and all Subscribers currently subscribed to the Channel have previously received a certificate from this Distributor, click No.
- ◆ If this is the first Distribution added to this Channel by the Distributor, or a Subscriber has been newly added to the Channel, click Yes (to resolve certificates).

This will copy the security certificates from the Distributor to the Subscribers that are subscribed to the Channel.

A prompt to copy a certificate is usually displayed when you have added:

- ◆ A Channel to a Distribution
- ◆ A Distribution to a Channel
- ◆ A Subscriber to a Channel
- ◆ A Channel to a Subscriber

To manually initiate resolving certificates:

- 1** In ConsoleOne, right-click the Distributor object > click Resolve Certificates.
- 2** Make sure the Copy Certificates Automatically to Subscribers radio button is checked > click OK.

This will copy the new certificate to each Subscriber so that it can receive Distributions from this Distributor, as long as the workstation where you are running ConsoleOne can contact all of the Subscriber servers. If you are prompted for a location to copy the certificates, you must have a drive mapped to the destination server.

Handling Invalid Certificates

A Subscriber cannot receive Distributions from a Distributor when the Distributor's certificate has become invalid. A Subscriber cannot receive encrypted Distributions when the Subscriber's encryption certificate has become invalid. For information on encryption certificates, see [“Distribution Security Using Encryption” on page 558](#).

A Distributor's certificate can become invalid when the DNS name or IP address of the Distributor has been changed. However, if your Distributor is configured to use DNS (the recommended addressing method), IP address changes on the Distributor will not invalidate its certificate. Also, if DNS addressing is being used, changes in a Subscriber's DNS name or IP address will not prevent the Subscriber from receiving Distributions.

However, a Subscriber's encryption certificate can become invalid when the DNS name or IP address of the Subscriber is changed, in which case a new encryption certificate needs to be created.

The following applies for DNS name changes where DNS is your installed addressing method, or for IP address changes where IP address is your installed addressing method:

- ◆ [“Distributor DNS Name or IP Address Is Changed” on page 554](#)
- ◆ [“Subscriber DNS Name or IP Address Is Changed” on page 555](#)

Distributor DNS Name or IP Address Is Changed

Because the Distributor identifies itself to Subscribers by its server's DNS name or IP address, if you change the identifier being used on the Distributor server, Subscribers will not recognize the Distributor as a valid source for Distributions.

Changing the DNS name or IP address of a Distributor causes the certificate created by the Distributor to be invalid for all Subscribers that have received the certificate from this Distributor. Therefore, the Distributor must send new certificates to all Subscribers receiving Distributions from that Distributor.

To re-create and resolve the Distributor's certificate, do the following in order:

- ◆ [“Modify the Distributor Server's Identification Attributes” on page 554](#)
- ◆ [“Create and Send New Certificates” on page 555](#)

Modify the Distributor Server's Identification Attributes

You must first modify the Network Address attribute on the Other tab in the Distributor and Subscriber objects' properties.

If the server is using the DNS Name attribute to identify itself, do the following:

- 1** In ConsoleOne, right-click the Distributor object > click Properties > click the Other tab.
- 2** Click the + symbol to the left of the NetWork Address.
- 3** Click the icon to the left of the field you want to modify.
A Browse button will be displayed to the right.
- 4** Click the Browse button.
- 5** If you are modifying the DNS Name field, click the drop-down list at the top of the box where Type 13 is displayed.
- 6** Change the value from Type 13 to IP > then change IP back to Type 13.
This resets the value to now recognize the new DNS name.
- 7** Click the Browse button to the right of the NetAddress field in the lower portion of the box.
- 8** Click Servers DNS Name (on the right side of the box) > change it to the new name.
- 9** Click OK to return to the Other tab.
- 10** Click OK to finish.

If the server is using the IP Address attribute to identify itself, do the following:

- 1** In ConsoleOne, right-click the Distributor object > click Properties > click the Other tab.
- 2** Click the + symbol to the left of the NetWork Address.
- 3** Click the icon to the left of the field you want to modify.
A Browse button will be displayed to the right.
- 4** Click the Browse button.
The IP address will be displayed in the lower portion of the dialog box.
- 5** Change the IP address to the new one.
- 6** Click OK to return to the Other tab.

7 Click OK to finish.

Continue with [“Create and Send New Certificates”](#) on page 555.

Create and Send New Certificates

1 On the Distributor server, shut down the Distributor Agent:

NetWare: At the ZFS console prompt, enter EXITALL.

Windows: In the Services dialog, select to stop each of the ZFS services.

For information on stopping and starting agents, see [“Starting the Policy and Distribution Services Agents”](#) in [“Installing on NetWare and Windows Servers”](#) in [“Installing Policy and Distribution Services on NetWare and Windows Servers”](#) in the *Installation* guide; or, see [“Starting the Policy and Distribution Agents on Linux or Solaris”](#) and [“Stopping the Policy and Distribution Services Agents on Linux or Solaris”](#) in [“Installing Policy and Distribution Services on Linux or Solaris Servers”](#) in the *Installation* guide.

2 In the ZENWORKS\PDS\TED\SECURITY\PRIVATE directory on the Distributor server, delete the .KEYSTORE file.

This file contains the Distributor’s certificate.

3 In the ZENWORKS\PDS\TED\SECURITY\CSR directory on the Distributor server, delete the .CSR file that has a name that matches either the old DNS name or the old IP address.

4 Restart the Distributor Agent.

A new certificate and .KEYSTORE file will be automatically created for the Distributor.

5 To send new certificates to all Subscriber that receive Distributions from the Distributor selected in [Step 1](#):

5a To resolve certificates, in ConsoleOne, right-click the Distributor object > click Resolve Certificates.

IMPORTANT: ConsoleOne copies the certificate files to Subscriber servers. Therefore, the client software on the workstation running ConsoleOne must have access to the Subscriber servers’ file systems. For Windows Subscriber servers, the Domain and Workgroup rights on the workstation must be set up to facilitate automatic certificate copying. Otherwise, a 1204a error will be given.

5b Make sure the Copy Certificates Automatically to Subscribers radio button is checked > click OK.

This will copy the new certificate to each Subscriber so that it can receive Distributions from this Distributor, as long as the workstation where you are running ConsoleOne can contact all of the Subscriber servers. If you are prompted for a location to copy the certificates, you must have a drive mapped to the destination server.

Subscriber DNS Name or IP Address Is Changed

Because the Distributor obtains the address of a Subscriber from the Subscriber’s object in eDirectory, this information must be updated in the Subscriber object so that it can receive its Distributions.

Changing the DNS name or IP address of a Subscriber causes all encryption certificates contained on the Subscriber to be invalid. Subscribers can have one encryption certificate from each Distributor that sends it encrypted Distributions.

Subscribers can continue to receive non-encrypted Distributions, even if the DNS name or IP address is changed.

The following sections outline the steps to resolve DNS name or IP address changes:

- ◆ [“Modify the Subscriber Server’s Identification Attributes” on page 556](#)
- ◆ [“Resolve the New Certificates” on page 556](#)

Modify the Subscriber Server’s Identification Attributes

You must first modify the Network Address attribute on the Other tab in the Distributor and Subscriber objects’ properties. To accomplish this, do the following as applicable.

If the server is using the DNS Name attribute to identify itself, do the following:

- 1** In ConsoleOne, right-click the Subscriber object > click Properties > click the Other tab.
- 2** Click the + symbol to the left of the NetWork Address.
- 3** Click the icon to the left of the field you want to modify.
A Browse button will be displayed to the right.
- 4** Click the Browse button.
- 5** If you are modifying the DNS Name field, click the drop-down list at the top of the box where Type 13 is displayed.
- 6** Change the value from Type 13 to IP > then change IP back to Type 13.
This resets the value to now recognize the new DNS name.
- 7** Click the Browse button to the right of the NetAddress field in the lower portion of the box.
- 8** Click Servers DNS Name (on the right side of the box) > change it to the new name.
- 9** Click OK to return to the Other tab.
- 10** Click OK to finish.

If the server is using the IP Address attribute to identify itself, do the following:

- 1** In ConsoleOne, right-click the Subscriber object > click Properties > click the Other tab.
- 2** Click the + symbol to the left of the NetWork Address.
- 3** Click the icon to the left of the field you want to modify.
A Browse button will be displayed to the right.
- 4** Click the Browse button.
The IP address will be displayed in the lower portion of the dialog box.
- 5** Change the IP address to the new one.
- 6** Click OK to return to the Other tab.
- 7** Click OK to finish.

Resolve the New Certificates

To reproduce valid encryption certificates for the Subscriber, follow the instructions under [“Distribution Security Using Encryption” on page 558](#).

Certificate and Private Key Directories

Certificates and private keys for Policy and Distribution Services are stored in the following locations in the .KEYSTORE file:

- ◆ For the Distributor's private key on a NetWare Distributor server:

```
SYS:\ZENWORKS\PDS\TED\SECURITY\PRIVATE
```

- ◆ For the Distributor's private key on a Windows Subscriber server:

```
C:\ZENWORKS\PDS\TED\SECURITY\PRIVATE
```

- ◆ For certificates received from Distributors on a NetWare Subscriber server:

```
SYS:\ZENWORKS\PDS\TED\SECURITY
```

After the Distribution has been sent, the certificate is moved into the .KEYSTORE file.

Creating Security Certificates for Non-Encrypted Distributions

To create a certificate on a Distributor and copy it to its associated Subscribers:

- 1** On the server where a Distributor is installed, make sure its Distributor Agent is running (use TED.NCF on a NetWare server, restart the Novell ZfS Distribution service on a Windows server, or enter `/etc/init.d/zfs start` on a UNIX server).

This Java process will create the certificate and write it into eDirectory.

- 2** Copy the certificate to each Subscriber using one of the following methods:

- ◆ If your Channels and Distributions are set up, in ConsoleOne, right-click the Distributor object > click Resolve Certificates > click OK. Make sure the Copy Certificates Automatically to Subscribers radio button is checked before clicking OK. This will copy the new certificate to each Subscriber so that it can receive Distributions from this Distributor.

For information on resolving certificates, see [“Resolving Certificates” on page 552](#).

- ◆ If necessary, associate Subscribers with a Channel > create a Distribution for the Distributor > associate the Distribution with a Channel. When you click OK you will be prompted to resolve the certificate. Respond to the query with Yes to resolve certificates for all Subscribers. The certificates are copied to all of the associated Subscribers. The Subscriber Java process does not need to be running on the Subscriber server; the server only needs to be up.
- ◆ Manually copy the Distributor's certificate to each Subscriber server's `installation_path\ZENWORKS\PDS\TED\SECURITY` directory (on UNIX, `usr/ZENworks/PDS/TED/Security`).
- ◆ Right-click a Subscriber object > click Resolve Certificates (repeat for each Subscriber object). This option might only be available if you answered No when prompted to copy security certificates.

Note that the first two options are the easiest when there are many Subscribers receiving Distributions from one Distributor.

- 3** Because each Distributor creates its own security certificate, repeat [Step 1](#) and [Step 2](#) for each Distributor object in the tree.

Manually Copying Certificates for Non-Encrypted Distributions

To manually copy certificates to Subscribers using ConsoleOne, do the following:

- 1 Right-click a Distributor, Subscriber, or External Subscriber object > click Resolve Certificates.

or

Click File > Resolve Certificates.

- 2 Click the Save Certificates to Disk radio button.
- 3 Enter a path for where to copy the certificate file > click OK.

The certificate file that is copied to this path will be named using the following syntax:

DNS_Name.CER

- 4 Copy the *DNS_name.CER* file from the path you gave to the Subscriber server's ZENWORKS\PDS\TED\SECURITY directory (on UNIX, `usr/ZENworks/PDS/TED/Security`).

Distribution Security Using Encryption

Policy and Distribution Services provides the option to encrypt a Distribution to prevent unauthorized access to its contents when the Distribution is sent outside your secured network. There is usually no need to encrypt Distributions that are sent within your secured network.

Encrypting Distributions is basically a two-step process:

1. Click the Encrypt check box in the Distribution's properties in ConsoleOne and select the level of encryption (strong or weak).
2. Manually create and copy the encryption security certificate files between the Distributor and Subscriber servers.

IMPORTANT: For security, you should use a physical medium, such as a diskette, to transfer the certificate between network servers.

Thereafter, the Distribution will be sent as an encrypted Distribution.

To understand Distribution encryption, review the following:

- ◆ [“Creating and Copying Encryption Certificates” on page 558](#)
- ◆ [“Sending an Encrypted Distribution” on page 560](#)
- ◆ [“Extracting an Encrypted Distribution” on page 561](#)

Creating and Copying Encryption Certificates

RSA PKIs provide the security process used for encrypted TED Distributions.

Encryption certificates are created from Certificate Signing Request (.CSR) files. Every Subscriber server contains a .CSR file that can be used as a template for creating an encryption certificate for a particular Distributor.

The encryption certificates (.CER) are used by the Subscribers to ensure secure transmission of an encrypted Distribution. If you pass the .CER file over the wire, the Distribution's encryption key

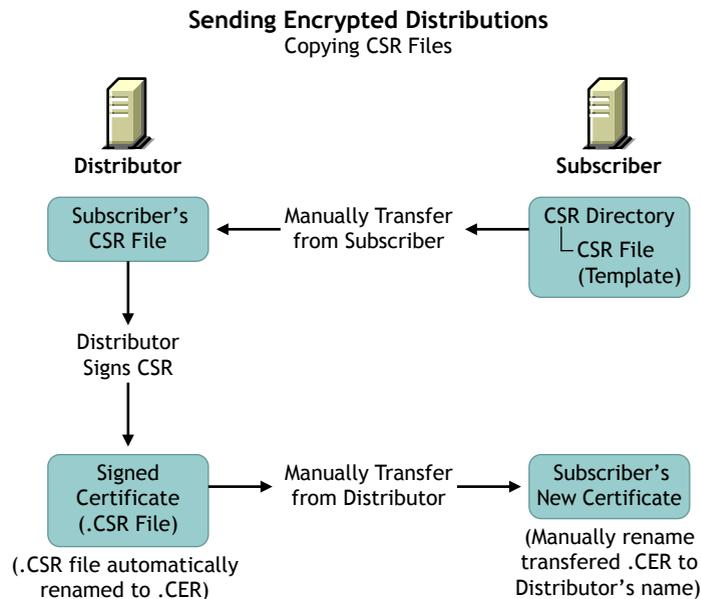
could be compromised. Therefore, you must manually copy the encryption security certificates to ensure that the encryption key contained in the certificate files are kept secure.

IMPORTANT: Do not manually copy a certificate by using a file browser, because that uses transmission lines and can be compromised. Instead, copy the certificate to an external media, such as a floppy diskette, and transport it physically between the Distributor and Subscriber servers.

To use encryption certificates with Subscribers, you must have previously resolved certificates and sent a non-encrypted Distribution to each Subscriber.

For information on resolving certificates, see [“Resolving Certificates” on page 552](#).

The following illustrates the process of manually copying the encryption certificates:



The Distributor signs the .CSR to create the encryption .CER file, which is manually copied from the Distributor to the Subscriber to replace the current non-encryption .CER file on the Subscriber server.

The encryption certificate is required for extracting a Distribution. If a Subscriber is only acting as a parent Subscriber to pass the encrypted Distribution on to Subscribers who have subscribed to the Distribution's Channel, the parent Subscriber does not need to have the encryption certificate on its server.

To create certificates for an encrypted Distribution:

- 1** Determine the Distribution you want encrypted.
- 2** Determine the Distributor that owns this Distribution.
- 3** Determine which Subscribers will be receiving the encrypted Distribution.
- 4** Resolve certificates for the selected Distributor to the selected Subscribers > send a non-encrypted Distribution from that Distributor to the Subscribers.

For information on resolving certificates, see [“Resolving Certificates” on page 552](#).

- 5** Access the file systems of this Distributor and these Subscribers.
- 6** Copy every .CSR certificate file contained in the following directory from each Subscriber to the same path on the Distributor:

`\ZENWORKS\PDS\TED\SECURITY\CSR`

This path begins with whatever you used for installing ZfS.

The Certificate Signing Request (.CSR) is used to create the encryption certificate file.

- 7** In ConsoleOne, right-click the Distributor object > click Sign CSR Files > select the .CSR files to be signed > click Sign > click OK on the Success dialog box > click Close.

You can select multiple .CSR files to be signed at the same time.

This creates the Certificate (.CER) files in the same Distributor's directory as the .CSR files you copied from the Subscribers. You will have one .CER file for each .CSR file.

You can also perform this step using Novell iManager:

7a Click Remote Web Console.

7b Select or enter the Distributor's IP address.

7c In the Available Services drop-down box, select Tiered Electronic Distribution.

7d Select the Security tab > click the Sign CSR link.

- 8** For each target Subscriber, do the following:

8a Copy the Subscriber server's corresponding .CER files from the following location on the Distributor's file system:

`\ZENWORKS\PDS\TED\SECURITY\CSR`

to the following path on the Subscriber's own server's file system:

`\ZENWORKS\PDS\TED\SECURITY`

TIP: Each .CER file contains its Subscriber server's name.

8b Rename the .CER files that you just copied to the Subscriber server to have the Distributor's DNS name instead of the Subscriber's.

- 9** Send the encrypted Distribution.

WARNING: Under the following scenario, the encryption certificates you just created can be overwritten before they are used:

1. Changes are made to the Channel, Subscribers, or Distribution involved with the encrypted Distribution.
2. This causes the prompt for copying certificates to be displayed.
3. If you reply with Yes before the encrypted Distribution has been sent and received by the Subscribers:
 - a. The encryption .CER file will be overwritten on each Subscriber with a non-encryption .CER file.
 - b. The Subscribers will not be able to decrypt the Distribution when it is received, because the .CER file was overwritten with a .CER file that does not contain the encryption keys.

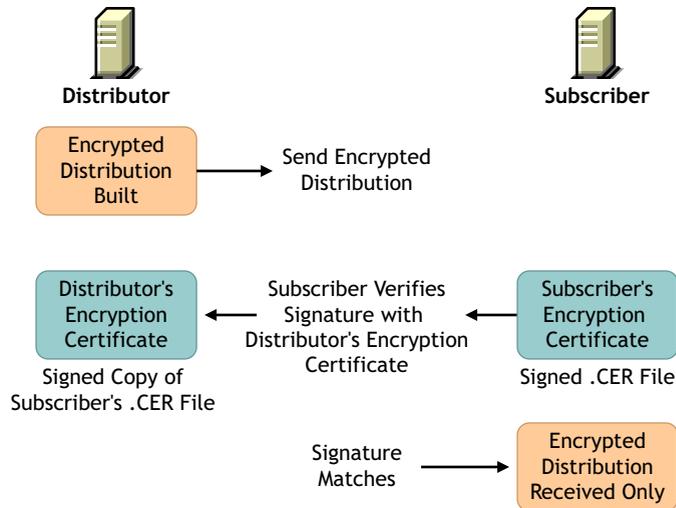
After the encrypted Distribution has been sent once to each Subscriber, the encryption .CER file is moved into the .KEYSTORE file on the Subscriber server's file system so that it cannot be overwritten. Thereafter, you can reply with Yes to copy certificates when this scenario occurs.

Sending an Encrypted Distribution

After an encryption certificate has been established on a Subscriber server, the following illustrates the process for sending encrypted Distributions:

Sending Encrypted Distributions

Sending the Distribution



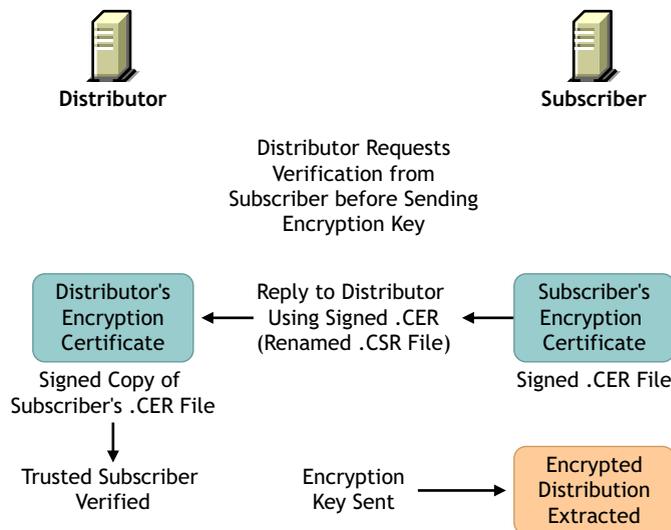
The only Subscribers that need to receive the encryption key are those that will be extracting the Distribution. Therefore, parent Subscribers and Subscribers in the Distributor's routing hierarchy do not need to receive the encryption key if they will not be extracting the Distribution.

Extracting an Encrypted Distribution

Before an encrypted Distribution can be extracted on a Subscriber server, the Subscriber must receive the encryption key. The following illustrates how the key is sent:

Sending Encrypted Distributions

Sending the Encryption Key



Each Distribution has its own encryption key sent.

Security for Inter-Server Communication Across Non-Secured Connections

Policy and Distribution Services uses XMLRPC (Extensible Markup Language Remote Procedure Call) for its normal inter-server communications. XMLRPC optionally provides security for inter-server communication across non-secured connections. Policy and Distribution Services can use this security for inter-server communications between servers across non-secured connections, or between a management workstation and servers across non-secured connections. For example, firewalls, intranets, NAT configurations, and so on.

This inter-server communications security ensures that data received across a non-secured connection is from a trusted source, that it has not been tampered with en route, and that the data received can be trusted by other machines. This is accomplished through the use of signed security certificates and digital signatures.

This security requires modifications to certain text files, and is installed using a ZfS wizard.

For instructions on installing XMLRPC security, see [“Installing Additional Security for Non-Secured Connections”](#) in the *Installation* guide.

The following are instances when you would want inter-server communication security:

- ◆ **ConsoleOne Administration:** When you use a workstation to manage a Distributor server across a non-secured connection.
- ◆ **SET Parameters:** When you create a SET Parameter policy or a software package for SET parameters, inter-server communication takes place to provide the target server’s SET parameter information. This communication could cross a non-secured connection.
- ◆ **Server Down Policy:** When you use this policy to down a server, the communication between the downed server and another server watching for it to come back up could cross a non-secured connection.

Review the following sections to understand inter-server communications security using XMLRPC:

- ◆ [“Terms Used in This Section” on page 562](#)
- ◆ [“Security Certificates” on page 563](#)
- ◆ [“Using SSL for the ZenCSServlet” on page 563](#)
- ◆ [“Format of the Password File” on page 564](#)

Terms Used in This Section

The following terms and acronyms are used in the inter-server communications security documentation:

Term	Explanation
CA	Certificate Authority The trusted certificate source responsible for digitally signing other server’s X.509 certificates.

Term	Explanation
CS	Certificate Signer The trusted certificate source responsible for digitally signing other server's XMLRPC certificates.
certificate or security certificate	An electronic document that contains an electronic signature for validating anything associated with the certificate, such as a Distribution.
CSR	Certificate Signing Request Request by a server to have an XMLRPC certificate signed by the trusted CS. This is not an X.509 certificate that would be signed by a root CA, such as VeriSign or Thawte Consulting.
self-signed certificate	A valid certificate signed by its creator.
signed certificate	A certificate signed by a CS, which makes it valid for acceptance by the receiving server.
SSL	Secure Socket Layer
XMLRPC	Extensible Markup Language Remote Procedure Call Software used by ZfS and TED for inter-server communications.
ZenCSServlet	ZENworks Certificate Signer Servlet Servlet that implements the Certificate Signer functionality.

Security Certificates

Inter-server communications security uses signed certificates issued by the Certificate Signer (CS), which are valid only within the context of the Novell ZENworks family of products.

The certificates used are not X.509 compliant and cannot be used for any e-commerce or SSL applications.

However, because SSL can be used by inter-server communications security for the ZenCSServlet, X.509 certificates provided by SSL can be used to secure inter-server communications. These certificates could be self-signed or signed by a root CA, such as VeriSign.

Using SSL for the ZenCSServlet

When a CS servlet signs a Certificate Signing Request (CSR), the requesting client must authenticate with a username and password via HTTP Basic Authentication. You can secure the username and password by using SSL for the ZenCSServlet.

For information on how to enable SSL for a commercial Web server, see your SSL documentation.

For information on setting up SSL with the ZfS Web Server, see [“Configuring the Zen Web Server to Use SSL”](#) in [“Configuring Other Related Components”](#) in [“Installing Additional Security for Non-Secured Connections”](#) in the *Installation* guide.

Format of the Password File

Inter-server communications security uses a password file for the username and password that are authenticated for CSR signing. You can create the password file in a text editor and place it in any secure location. You should also restrict access to the file to only the users who are listed in the file.

Usernames and passwords are both case sensitive. The syntax for the password file is:

username=password

For example:

```
admin=adminpassword  
CSsigner=cspassword  
JohnDoe=jdpassword
```

You should limit the access to the password file to those users included within the file.

TCP/IP Addresses and DNS Names

In setting up inter-server communications security, the installation program relies on addresses or names of the servers where you want this security enabled. You can use either TCP/IP addresses or fully distinguished DNS server names.

IMPORTANT: For NetWare servers, DNS names cannot have underscores. Distribution sending or receiving errors will occur if the server's DNS name contains underscores. We recommend that you use dashes instead of underscores as word separators.

For the various methods you can use to obtain these addresses or server names, see [“Information to Know Before Beginning the Installation”](#) in [“Installing Inter-Server Communications Security”](#) in [“Installing Additional Security for Non-Secured Connections”](#) in the *Installation* guide.

21 Scheduling

The following information on scheduling applies to Novell® ZENworks® for Servers (ZfS) Policy and Distribution Services:

- ◆ [“Understanding Scheduling in Policy and Distribution Services” on page 565](#)
- ◆ [“Scheduling Issues” on page 565](#)
- ◆ [“The Schedule Types” on page 570](#)
- ◆ [“Scheduling Server Policies” on page 573](#)
- ◆ [“Scheduling the TED Objects” on page 574](#)
- ◆ [“Using Intervals and Repeating Actions in Schedule Types” on page 577](#)

Understanding Scheduling in Policy and Distribution Services

When you schedule a server policy or a Tiered Electronic Distribution (TED) Distribution, you can select its type of schedule. The type of schedule you select depends on which policy it is or what is contained in the Distribution.

For information the types of schedules, see [“The Schedule Types” on page 570](#).

You can also define a window of opportunity during the day for when a schedule’s action is to begin and end. Distributions are anticipated to occur during off-peak hours. For some networks, it is possible that the scheduling window can be very short. Other systems on the network also use off-peak hours for processing, such as backups.

You can have instances where the limiting factor is available time; therefore, the critical condition is how fast the distributions can take place, regardless of the resources consumed. You might need to experiment to determine the best relationship between time and resources.

Scheduling dictates when network resources are used by a Distribution.

For an understanding of some issues related to scheduling in Policy and Distribution Services, see [“Scheduling Issues” on page 565](#).

Scheduling Issues

The following explain various scheduling issues:

- ◆ [“Scheduling Differences Between Policies and TED” on page 566](#)
- ◆ [“Scheduling Conflicts with Other Software” on page 566](#)
- ◆ [“Randomly Dispatch Option Issues” on page 566](#)
- ◆ [“Distributor Scheduling Issues” on page 568](#)

- ◆ [“TED Object Scheduling Issues” on page 568](#)
- ◆ [“Calculating Time Differences” on page 568](#)
- ◆ [“Inactivating Distributions and Channels” on page 569](#)

Scheduling Differences Between Policies and TED

Policies are scheduled according to local times. TED objects are scheduled according to an offset from Greenwich Mean Time (GMT).

Policies example: If you are residing in Utah and set a policy to be executed at 5 p.m. Utah time, it would be executed at 5 p.m. local time in Utah for servers residing in Utah. In California, it would execute at 5 p.m. local time in California. In other words, setting a time of 5 p.m. for a policy makes it execute at 5 p.m. local time wherever the servers reside.

TED example: If you are residing in Utah during Daylight Saving Time and set a TED object’s schedule for 5 p.m., it would be executed at 5 p.m. local time in Utah. In California, it would execute at 4 p.m. local time (5 p.m. in Utah) for servers residing in California. In other words, TED schedules are relative to a GMT offset that makes the TED schedule execute at the exact same moment worldwide.

Scheduling Conflicts with Other Software

Distributions are anticipated to occur during off-peak hours. For some networks, it is possible that this scheduling window could be very short. Other systems on the network can also use off-peak hours for processing, such as backups.

You might have instances where the limiting factor is available time; therefore, the critical condition is how fast the Distributions can take place, regardless of the resources consumed. You might need to experiment to determine the best relationship between time and resources.

Randomly Dispatch Option Issues

The Randomly Dispatch During Time Period option is available for each of the schedules (Distributor, Subscriber, Channel, and Distribution). It is used in conjunction with a time window (Start and End times) that you can set for a Daily, Monthly, or Yearly schedule type.

Randomly dispatching causes the scheduled action to run at any time during the window for the day. This helps load-balancing on servers. However, random-dispatched schedules can be confusing if you are expecting an action to take place immediately.

The following describe the issues for the Randomly Dispatch option:

- ◆ [“Using the Randomly Dispatch Option in a Distributor’s Refresh Schedule” on page 566](#)
- ◆ [“Using the Randomly Dispatch Option in a Distribution’s Build Schedule” on page 567](#)
- ◆ [“Using the Randomly Dispatch Option in a Channel’s Send Schedule” on page 567](#)
- ◆ [“Using the Randomly Dispatch Option in a Subscriber’s Extract Schedule” on page 568](#)

Using the Randomly Dispatch Option in a Distributor’s Refresh Schedule

You can use the Randomly Dispatch option for Distributor Refresh schedules to load balance Distributor refreshes from eDirectory. This is useful to minimize the network traffic that can be caused by many Distributors trying to read eDirectory at the same time.

Be sure to coordinate a Distributor's Refresh schedule with that Distributor's related Distributions' Build and Channels' Send schedules.

The Distributor's Refresh schedule should be determined by how frequently TED information is updated in eDirectory. For example, how often new Distributions are created, properties of existing Distribution objects changed, new Channels are added, and so on. The Distributor cannot know of changes made to TED objects without re-reading eDirectory. An eDirectory refresh should finish before the Build and Send schedules begin.

IMPORTANT: Do not refresh the Distributor more often than every five minutes. The following can need up to five minutes to complete their processes: Distribution building, eDirectory replication, and tree walking (when no Search policy is defined).

If you are using the Randomly Dispatch option, you should consider the End time for the Refresh schedule when setting the Start times for the Build and Send schedules.

Using the Randomly Dispatch Option in a Distribution's Build Schedule

You can use the Randomly Dispatch option for a Distribution's Build schedule to load-balance the Distributor's work in building Distributions. This becomes more necessary as the number of Distributions for a Distributor grows.

Be sure to coordinate a Distribution's Build schedule with its Distributor's Refresh schedule and any related Channels' Send schedules. A Distribution build should begin after the Refresh schedule ends and finish before the Send schedules begin.

IMPORTANT: Do not refresh the Distributor more often than every five minutes. The following can need up to five minutes to complete their processes: Distribution building, eDirectory replication, and tree walking (when no Search policy is defined).

If you are using the Randomly Dispatch option, you should consider the End time for its Distributor's Refresh schedule when setting the Build schedule's Start time; and, you should consider the End time for the Build schedule when setting the Start times for the Send schedules.

Using the Randomly Dispatch Option in a Channel's Send Schedule

You can use the Randomly Dispatch option for a Channel's Send schedule to begin sending its Distributions to Subscribers randomly within a scheduling window. Each Distributor that has Distributions in the Channel calculates a random time between the specified Start and End times to begin sending its Distributions. This helps to balance the distribution workload for the network over a period of time.

For example, Distributor A and Distributor B have Distributions in a Channel. Each Distributor would calculate its own random time to begin sending its Distributions.

Another use of the Randomly Dispatch option for the Send schedule is if you have many Channels and you want all Distributions for all Channels to occur between 10 p.m. and 4 a.m. Using the Randomly Dispatch option in each Channel would allow you to disperse Distribution sending times for all Channels over that six-hour period of time.

If you are using the Randomly Dispatch option, you should consider the End time of each associated Distribution's Build schedule when setting the Send schedule's Start time; and, you should consider the End time for the Send schedule when setting the Start times for all associated Subscribers' Extract schedules.

Using the Randomly Dispatch Option in a Subscriber's Extract Schedule

You can use the Randomly Dispatch option for a Subscriber's Extract schedule to balance the Subscriber's work load in extracting Distributions.

If you are using the Randomly Dispatch option, you should consider the End times for the Send schedules of the Channels where the Subscriber is subscribed when setting the Start time for the Extract schedule.

Distributor Scheduling Issues

Your Distributor can start sending Distributions to Subscribers throughout the scheduling window, according to the associated Channel schedules (see [“TED Object Scheduling Issues” on page 568](#)).

Use the Daily, Monthly, or Yearly schedule with the Randomly Dispatch option, in conjunction with the Maximum Number of Concurrent Distributions option, to help with load-balancing for Distributors. This spreads the network traffic that is caused by sending many Distributions over the entire scheduling window.

TED Object Scheduling Issues

The following information concerning time zone offsets is from the perspective of the Channel object. However, this information is applicable to all TED objects that can be scheduled.

Because a Channel is an object in the tree that is not associated with a specific server, the Channel's time is always set to the local time zone of the workstation that is running ConsoleOne® and setting the Channel's schedule.

For example, if you (the administrator) live in New York City, the local time for any Channels you schedule from there will be local New York time.

If Distributors in different time zones from the Channel have Distributions in that Channel, the Distributors will need to send their Distributions according to the Channel's local time schedule. For example:

1. You set a Channel's schedule to be from 1 a.m. through 5 a.m. local time in Los Angeles.
2. In New York you select to have a Distributor's Distribution listed in that Los Angeles Channel.
3. The Distribution can be sent only between 4 a.m. and 8 a.m. in New York because for New York, being three hours ahead of Los Angeles, its time window of 4–8 a.m. is happening at the same time as the Los Angeles time window of 1–5 a.m.

You should use a time zone offset to determine the true local time when the Distributor can send its Distributions. Also, because a Channel's schedule determines when a Distribution can be sent, you must make sure the build schedules you set for your Distributions will occur before a Channel's schedule.

Calculating Time Differences

The [World Time Server \(http://www.worldtimeserver.com\)](http://www.worldtimeserver.com) is a Web site where you can determine the time difference between any two locations in the world.

As you look at the site, note the following:

- ◆ The locations in the left frame can be listed by countries or major cities.

- ◆ The current GMT time relative to the International Date Line is displayed in the right frame.
- ◆ When you click a location in the left frame, the time displayed in the right frame includes the day, date, whether Standard Time or Daylight Saving Time is in effect, and the GMT offset.

To use this site to calculate time differences between TED locations,

- 1** Click the location for one of the TED sites.
- 2** Note the time, day/date, GMT offset, and whether Daylight Saving Time is in effect (for future reference).
- 3** Click the location for another TED site.
- 4** Note the time, day/date, GMT offset, and whether Daylight Saving Time is in effect.
- 5** Repeat this process for all of the TED locations where you want to coordinate schedules.
- 6** Using the information you have gathered, calculate the time differences between the TED locations.
- 7** Taking into consideration when events will be taking place locally at the various TED locations, configure the appropriate schedules using the time differences.

As an example,

- ◆ A Distributor in Hawaii lists a Distribution in a Channel in New York.
- ◆ Using the World Time Server Web site, you will find that the offset between the two locations is -6 when Daylight Saving Time is in effect. (The negative number means it is later in the time sequence, so you must subtract Hawaii's time from New York's time to arrive at the correct a.m. or p.m.)
- ◆ If the Channel's starting time is 1 a.m. in New York, select 7 p.m. for the Distributor's schedule in Hawaii.
- ◆ The result is that the Distributor can start to send its Distribution at 7 p.m.
- ◆ Because Hawaii is not observing Daylight Saving Time and New York is, when New York moves back to Standard Time, the result would be 8 p.m.

If you wanted the Distributions to be sent later in the evening in Hawaii, the Channel's time window would have to start later than at 1 a.m. in New York. For example:

- ◆ You want the Distributions to begin sending at 11 p.m. in Hawaii.
- ◆ You need to set the Channel's start time to be 5 a.m. in New York.

When you set up your Channel schedules, you need to consider which object's time window is more important. For example, it might be more important for the Distributor to be sending Distributions during off-peak hours. Therefore, using the New York and Hawaii example, to have the Distributions begin sending after midnight Hawaii time, you would need to have the New York Channel's start time set to 6 a.m. or later.

Inactivating Distributions and Channels

A Distribution can be set as Active or Inactive. Inactive is used when you are building a Distribution because you want to keep it inactive until it is ready to be sent to a Subscriber. The Active check box is found on the General tab of the Distribution object.

We recommend that your Channel be set to Inactive until you are ready to begin distributing your Distribution packages. This will prevent Distributions from being sent inadvertently.

The Schedule Types

The following table describes each of the schedule types.

Schedule Type	Description
Daily	Runs the scheduled item daily. Daily includes specifying a run time window, running randomly within the window of time, and running repeatedly every xxx hours or minutes. Used by all Policy and Distribution Services components.
Event	Runs the scheduled policy according to the specified event, such as at system startup or shutdown, or a third-party application-defined event. Used only by policies.
Interval	Repeats running the scheduled item every xxx days, hours, minutes, and/or seconds. For Distributors only, the interval begins after the Distributor re-reads eDirectory. Any frequency from a few seconds to many days can be specified. Used by policies, Distributors, Distributions, Channels, and Subscribers.
Monthly	Runs the scheduled item on the selected day of the month. Monthly includes specifying a run time window and running randomly within the window of time. Used by all Policy and Distribution Services components.
Never	Prevents a TED Distribution from running automatically. Used only by TED. Because the Distributor reads eDirectory but the Subscriber does not, the Subscriber will not know its working directory for a Distribution until its configuration is sent to it from the Distributor. Therefore, do not use the default of Never in any of the schedules, or you can receive an error that the Subscriber's working directory is unknown.
Package Schedule	Runs the scheduled item according to the default schedule, which can be changed on the Policies tab. Used only by policies.
Relative	Runs the scheduled policy one time relative to a specified number of days, hours, minutes, and seconds from when the policy package is extracted. For example, if you set the time to one hour and refresh the Distributor, a new policy package will be sent to the Subscriber, and it will run one hour after extraction. Used only by policies. Any time range, from a few seconds to many days, can be specified.
Run Immediately	Runs the scheduled item immediately upon refreshing the policy, beginning after the Distributor re-reads eDirectory. Includes repeating the action every xxx days, hours, minutes, and seconds. Any frequency from a few seconds to many days can be specified. Used only by policies, Distributions, Channels, and Subscribers.
Time	Runs the scheduled item once at the date and time specified. Used by all Policy and Distribution Services components.
Weekly	Runs the scheduled item on the selected day of the week. Weekly includes specifying a run time window, and running randomly within the window of time. Used only by policies.
Yearly	Runs the scheduled item on the selected day of the year. Yearly includes specifying a run time window, and running randomly within the window of time. Used by all Policy and Distribution Services components.

Daily

To schedule an item to run daily:

- 1 Click the down arrow on Schedule Type > select Daily > select one or more days of the week.
- 2 In Start Time, select the time the schedule will start for the day.
- 3 In End Time, select the latest time in the day for the schedule to run.
- 4 To have the schedule start randomly during the selected time period, check the Randomly Dispatch check box.
- 5 To have the schedule repeat the action, check the box for the Repeat the Action Every field > select how often the action should be repeated.

You can leave any of the options zeroed, but you must have a value in at least one of the time increments.

Event

To schedule a policy to run when an event happens:

- 1 Click the down arrow on Schedule Type > select Event > select which event will activate the schedule:

Event	Description
System Startup	Runs the action when the system starts up.
System Shutdown	Runs the action before the system shuts down.
Custom Event ID	Third-party application-defined event.

Interval

To schedule an item to run at an interval of time:

- 1 Click the down arrow on Schedule Type > select Interval > select the interval of time for repeating the action.

You can leave any of the options zeroed, but you must have a value in at least one of the time increments.

Monthly

To schedule an item to run monthly:

- 1 Click the down arrow on Schedule Type > select Monthly > click the radio button > select the day of the month.

or

Click the radio button for the last day of the month (whether 28, 29, 30, or 31).

- 2 In Start Time, select the time the schedule will start for the day.
- 3 In End Time, select the latest time in the day for the schedule to run.

- 4 To have the schedule start randomly during the selected time period, check the Randomly Dispatch check box.

Never

This type is only used by TED. It can be used to prevent a Distribution from running automatically.

To schedule a TED item to never run automatically:

- 1 Click the down arrow on Schedule Type.
- 2 Select Never.

Although you can specify to never run a Distribution, you can manually override this setting using the ZfS Management role in Novell iManager (see [“Forcing TED Agent Actions” on page 366](#)).

Package Schedule

Each policy package has a default schedule for all policies in that package.

You do not need to do anything to schedule a policy to run according to the current Default Package Schedule.

To change the Package Schedule:

- 1 In ConsoleOne, click the OU containing your server policies > right-click the Distributed Server Package (in the right pane) > click Properties.
- 2 Click Edit.
- 3 Change the Package Schedule to one of the following:
 - Daily
 - Weekly
 - Monthly
 - Yearly
 - Relative
 - Run Immediately
 - Event
 - Interval
 - Time

Relative

To schedule a policy to run relative to the time the policy package has been extracted:

- 1 Click the down arrow on Schedule Type > select Relative > select an amount of time.

You can leave any of the options zeroed, but you must have a value in at least one of the time increments.

Run Immediately

To schedule an item to run immediately:

- 1 Click the down arrow on Schedule Type > select Run Immediately.

2 If you want to repeat the action, click the Repeat check box.

3 Select a length of time.

You can leave any of the options zeroed, but you must have a value in at least one of the time increments.

Time

To schedule an item to run at a specific time:

1 Click the down arrow on Schedule Type > select Time > click the calendar icon.

2 In the Select Date and Time dialog box,

2a Select the month.

2b Select the year.

2c Click the day of the month.

2d Select the time of day > click OK.

Weekly

To schedule a policy to run weekly:

1 Click the down arrow on Schedule Type > select Weekly > select one day of the week.

2 In Start Time, select the time the schedule will start for the day.

3 In End Time, select the latest time in the day the schedule can run.

4 To have the schedule start randomly during the selected time period, check the Randomly Dispatch check box.

Yearly

To schedule an item to run yearly:

1 Click the down arrow on Schedule Type > select Yearly > click the calendar icon.

2 In the Select Date dialog box,

2a Select the month.

2b Click the day of the month.

3 In Start Time, select the time the schedule will start for the day.

4 In End Time, select the latest time in the day the schedule can run.

5 To have the schedule start randomly during the selected time period, check the Randomly Dispatch check box.

Scheduling Server Policies

Some policies must be scheduled before they can be enforced.

If you enable a policy, but do not schedule it, it will be activated according to the schedule currently specified in the Default Package Schedule.

The Default Package Schedule provides a default for scheduled policies. The default schedule is Run At System Startup.

The order of enforcement of different server policies is not guaranteed if the policies use exactly the same schedule. In other words, you should stagger the policies' schedules if you want to ensure the order they are enforced.

For information on scheduling policies, see [“Scheduling Policies” on page 488](#).

For information on policies, see [Chapter 17, “Server Policies,” on page 455](#).

Scheduling the TED Objects

TED uses schedules to control when Distributions are built, sent, and extracted. Schedules do not affect the total resources used by a Distribution (such as CPU cycles, bandwidth, and disk space), but rather when the resources will be used.

The following provides an understanding of scheduling in TED:

- ◆ [“Precedence of the Tiered Electronic Distribution Policy” on page 574](#)
- ◆ [“TED Object Schedules” on page 574](#)

Precedence of the Tiered Electronic Distribution Policy

If you set a schedule in the Schedule tab for the Tiered Electronic Distribution policy (in the Service Location Package), this schedule will be the default for all Distributors and Subscribers for which the policy applies, unless in ConsoleOne you set a schedule for a specific TED object. In other words, modified schedules for Distributors and Subscribers will automatically override the Tiered Electronic Distribution policy schedule.

The Distributor and Subscriber schedules are different. There are separate Schedule tabs for the Distributor's Refresh and Subscriber's Extract schedules.

By default, when a schedule is set in the Tiered Electronic Distribution policy, the Use Policy check boxes are displayed on both the General and Schedule tabs for all Distributors and Subscribers. And, the box is automatically checked for the Distributor and Subscriber objects that have not yet had their schedules modified. It is unchecked for the objects that have a schedule defined.

You can disable the Tiered Electronic Distribution policy's default schedule for a specific Distributor or Subscriber by unchecking the Use Policy check box in the object's properties. Then you must define a schedule in the object's properties for it to have a usable schedule.

You can override a specific Distributor or Subscriber schedule by checking the Use Policy check box in that object's properties. The Tiered Electronic Distribution policy's schedule will then be applied to that Distributor or Subscriber.

For information on how to create, configure, and schedule the Tiered Electronic Distribution policy, see [“Tiered Electronic Distribution” on page 471](#).

TED Object Schedules

Some TED objects must be scheduled or they will not perform their Distribution-related actions. TED has several schedules that are used to control when Distributions are built, distributed, and extracted.

You may or may not need to resolve certificates when making changes to one of the following schedules (for more information, see [“Resolving Certificates” on page 552](#)):

- ◆ [“Distributor Object’s Refresh Schedule” on page 575](#)
- ◆ [“Distribution Object’s Build Schedule” on page 575](#)
- ◆ [“Channel Object’s Send Schedule” on page 576](#)
- ◆ [“Subscriber Object’s Extract Schedule” on page 577](#)

Distributor Object’s Refresh Schedule

A Distributor’s schedule determines when the Distributor will re-read Novell eDirectory™ for configuration changes. This enables the Distributor to respond to a request to build a Distribution. The Distributor rebuilds a Distribution when the Distribution’s schedule indicates that it should be built.

When the Channel’s Send schedule starts, the Distributor checks with the Subscribers that it sends to directly to see if they have the current Distribution. However:

- ◆ If the Distribution is non-sequential, the Distributor simply checks for the current version.
- ◆ If the Distribution is sequential (the File or Desktop Application types of Distributions only), it checks to see if the Subscribers have all of the versions of the Distribution, starting with the baseline and every change since the baseline.

If the Subscriber does not have the entire Distribution, it checks with its subordinate Subscribers to see if they do, and so on down the routing hierarchy.

The time it takes to verify that all receivers have all of the Distributions in the Channel is minimal.

IMPORTANT: A Distribution might never get sent completely if the Refresh schedule is shorter than the time it takes to build or send the Distribution. In other words, if the Refresh schedule is too short, when the Distributor is refreshed the Distribution in the process of being built or sent could be cancelled before it has completed sending. Therefore, we recommend the Distributor’s Refresh schedule be daily, unless changes to Distributions warrant a more frequent refresh, then set it in hours. Do not refresh the Distributor more often than every five minutes.

Scheduling a Distributor

- 1** In ConsoleOne, right-click the Distributor object > click Properties.
- 2** Click the Schedule tab > click the arrow for the drop-down box > click Interval > select an interval, such as Daily.
- 3** Set the Start and End times, if necessary.

The Start Time and the End Time specify the time window for performing the schedule’s action.

You can repeat the action every so often throughout the day.

You can also have the refresh occur randomly in the specified time window. For more information, see [“Using the Randomly Dispatch Option in a Distributor’s Refresh Schedule” on page 566](#).

- 4** Click OK.

Distribution Object’s Build Schedule

The Distribution’s schedule determines when a Distributor will be requested to create the Distribution file based on the definition in the Distribution object.

Most Distributions consist of a set of files that change over time and need to be redistributed on a regular basis. Each Distribution has its own Build schedule that tells the Distributor how often to rebuild the Distribution. When the Distributor builds a Distribution, it automatically compares it with the previous version to see if there are any changes.

For the File Distribution, if there are no changes in the current build, no new version will be created. If there are changes, a delta is built consisting of only the changes to be distributed.

For the FTP, HTTP, and Software Package Distribution types, a new version will only be built if there has been a change since the last version. The Distributor will send the complete new version to all target Subscribers.

The Distribution's End Time is used to determine the end time for randomly dispatching events. In other words, the Distributor will not stop building the Distribution until it is complete.

Deleted files and directory synchronization are handled in the Build schedule.

Scheduling a Distribution

- 1** In ConsoleOne, right-click a Distribution object > click Properties.
- 2** Click the Schedule tab > click the arrow for the drop-down box > click a schedule type, such as Run Immediately.

You can repeat the action every so often.

The Start Time and the End Time specify the time window for performing the schedule's action.

You can also have the build occur randomly in the specified time window (if you select the Daily schedule type). For more information, see [“Using the Randomly Dispatch Option in a Distribution's Build Schedule” on page 567](#).

- 3** Click OK.

Channel Object's Send Schedule

A Channel's Send schedule provides a window of time for when a Distributor can start sending its Distributions to the Subscribers associated with that Channel.

The Channel's schedule applies only to the Distributor and its direct receivers (first tier Subscribers). When the Send schedule ends, the Distributor stops distributing to those first tier Subscribers.

Second-tier receivers and beyond do not adhere to the Channel's schedule. The parent Subscribers that are sending Distributions to other Subscribers will continue to send a Distribution after the Send schedule ends. Their subordinate Subscribers will also ignore the Send schedule.

The Send schedule's End Time forces the Distributor to stop sending a Distribution when the Send schedule ends. The Distributor will start sending the Distribution where it left off when the Send schedule begins again. A Distribution will not be totally re-sent. For example, if 50 MB of a 60 MB Distribution had already been sent before the disruption, when the Send schedule starts again for the Channel, the Distributor will begin sending the remaining 10 MBs.

For information on how time zones affect a Channel's schedule, see [“TED Object Scheduling Issues” on page 568](#).

Cache and Forward has no bearing on whether a parent Subscriber continues to send a Distribution when the Channel's Send schedule ends. Parent Subscribers who have completely received a Distribution prior to the Send schedule ending will continue to send that Distribution to

subordinate Subscribers. There is no mechanism for controlling whether parent Subscribers should continue to send when the Send schedule ends.

IMPORTANT: A Distribution might never get sent if the Send schedule is shorter than the time it takes to send the Distribution. Therefore, we recommend the Channel's Send schedule be daily or in hours. Make the Send schedule at least long enough to allow all of the Channel's Distributions to be sent.

Scheduling a Channel

- 1 In ConsoleOne, right-click the Channel object > click Properties.
- 2 Click the Schedule tab > click the arrow for the drop-down box > click Interval > select an interval (in the Repeat the Action Every field), such as 1 hour > click OK.

The Start Time and the End Time specify the time window for performing the schedule's action.

For information about randomly starting the Send schedule (if you select the Daily schedule type), see [“Using the Randomly Dispatch Option in a Channel's Send Schedule” on page 567](#).

Subscriber Object's Extract Schedule

The Subscriber's schedule determines when a Subscriber can extract a Distribution that has been received.

The Subscriber's End Time is used to determine the end time for randomly dispatching events. In other words, the Subscriber will not stop extracting the Distribution until it has completed the extraction process.

Scheduling a Subscriber

- 1 In ConsoleOne, right-click a Subscriber object > click Properties.
- 2 Click the Channels tab > click Add > browse for the Channel > click Select > click OK.
Make sure the Channel is listed as Active in the Channels list.
- 3 Click the Schedule tab > the arrow for the drop-down box > select a schedule, such as Run Immediately > Click OK.

This schedule type will cause the Subscriber to extract the Distribution as soon as it is received.

The Start Time and the End Time specify the time window for performing the schedule's action.

For information about randomly starting the Extract schedule (if you select the Daily schedule type), see [“Using the Randomly Dispatch Option in a Subscriber's Extract Schedule” on page 568](#).

- 4 Repeat these steps for each Subscriber.

Using Intervals and Repeating Actions in Schedule Types

Some scheduling options are common to several schedule types. These options must be understood before you can effectively use them.

- ♦ [“Using Intervals with Distributors” on page 578](#)
- ♦ [“Repeating Actions” on page 578](#)

Using Intervals with Distributors

For any schedule type that has an interval, the event will not start until after the Distributor has re-read eDirectory. For example:

- ◆ **Daily:** If the Distributor is refreshed before the current day's time window has passed, the event will run on the current day, then every day thereafter; otherwise, it will first run during that time window on the next day, then every day thereafter.
- ◆ **Interval:** If you set the interval to be three days, the event runs three days after the day the Distributor re-reads eDirectory, then runs every three days thereafter.
- ◆ **Weekly, Monthly, or Yearly:** The event runs the first day, month, or specific date (the Yearly option) after the Distributor has re-read eDirectory. For example, on Wednesday you set up a Weekly event to happen each Sunday. The Distributor re-reads eDirectory on Thursday, so the event runs the following Sunday, and every Sunday thereafter.
- ◆ **Run Immediately:** As soon as the Distributor is refreshed, the event runs, then runs thereafter according to the interval you set.

To cause an event for one of the interval-related schedule types to execute out of sequence (other than Run Immediately), you can use the ZfS Management role in iManager. For more information, see [Chapter 15, "Novell iManager," on page 359](#).

Repeating Actions

For schedule types that have the Repeat the Action Every field, how this option works depends on other factors, such as other schedules and how often the Distributor re-reads eDirectory.

For example:

- ◆ You select Daily as the Send schedule for a Channel
- ◆ You set 1:00 a.m. to midnight (23 hours) as the sending window
- ◆ You set the Repeat the Action Every field with 1 hour as the repeat value

The action (sending the Distribution) will repeat as follows:

1. Starting at 1:00 a.m. and repeating every hour, the Distributor will queue the Distribution to be sent.
2. If a Distribution is in the process of being sent, it will continue to be sent.
3. Once a Distribution is off the queue after being sent, the Distributor will queue the next newer version for sending.

If a previously queued version of this Distribution has not been sent yet, the next newest version will be placed in the queue. In other words, only one version of the Distribution (the last built) will be queued while another version of the Distribution is being sent. The Distributor always sends the latest Distribution, even if the Subscriber already has it.

22 Variables

This section is referenced from other sections.

You can use variables in Novell® ZENworks® for Servers (ZfS) to save time. For example, to globally control changes to the same location on all servers, versus making those changes manually in each script, you can use a variable for server names in the scripts. Then, when you want each script to recognize a new server name, just edit the variable.

Review the following sections for more information:

- ◆ [“Understanding Variables” on page 579](#)
- ◆ [“Types of Variables” on page 580](#)
- ◆ [“Resolution of Variable Names” on page 581](#)
- ◆ [“Nested Variables” on page 582](#)
- ◆ [“Creating a Variable” on page 582](#)
- ◆ [“Using Variables to Control File Extraction” on page 584](#)

Understanding Variables

Variables are used to simplify referencing information that is specific to individual servers. For example:

Server Name
Destination Volume
Working Directory
DNS Name
IPAddress

Variables can also be used to specify where a Distribution is to be extracted, including the full path. See the following sections for more information:

- ◆ [“Distribution Variable Example” on page 579](#)
- ◆ [“Where Variables Can Be Used” on page 580](#)

Distribution Variable Example

You have a single Distribution with 20 Subscribers. You want to extract the Distribution to a specific volume on each of the Subscriber’s servers. However, the volume name varies from server to server: 15 servers are using the DATA volume and five are using VOL1.

The Distribution Volume variable can be edited for some of these Subscribers by changing the Resolve To field on the Subscriber from DATA to VOL1 for the five Subscribers using that volume.

When the Distribution is extracted, it will go to the correct volumes on each of the 20 servers.

Where Variables Can Be Used

Examples of where you can use variables for Distribution packages include:

- ◆ Destination volume for a Distribution
- ◆ Destination directory for a Distribution

For the Server Software Package component, variables can be used anywhere you can enter text. For example:

- ◆ The name of text files that will be modified
- ◆ Within the modifications made to text files
- ◆ The content of a script to be run
- ◆ The directory where the software package will be installed

Types of Variables

There are two types of variables:

- ◆ “Predefined Variables” on page 580
- ◆ “User-Defined Variables” on page 581

Predefined Variables

Predefined variables are created when ZfS starts. They are used in software packages and are recognized by policy packages.

Predefined variables are not case sensitive, although they are displayed in all uppercase on the server console and in this documentation.

Syntax:

%predefined_variable_name%

where *predefined_variable_name* is the name defined by ZfS. For example:

`%WORKING_PATH%`

To make a predefined variable useful, its value must be set in the Server Software Package component.

The Java environment can use predefined variables, such as SERVER_DN being used in a Java process call in an .NCF file.

An example of how a policy package can use a predefined variable is for the Broadcast Message text in the Server Down Process policy. The text can include a variable for the server name (%SERVER_DN%) so that the broadcast message will display the name of the server.

The following ZfS predefined variables are available:

Variable	Description
BASE_PATH	Location of the Policy Manager (for example, SYS:\ZENWORKS\IPDS\SMANAGER).
IP_ADDRESS	IP address of a server.
LOAD_DIR	(NetWare® only) Directory where the server was loaded from.
PLUGINS_PATH	Path where the ZfS plug-ins were installed.
POLICY_PATH	Path where the policy files (.POL) are stored.
PROP_PATH	Path where Novell eDirectory™ object properties are stored.
SERVER_DN	Distinguished server name in eDirectory.
SERVER_NAME	Name given the server when NetWare was installed.
TED_PATH	Path to the TED directory.
TREE_NAME	Name of the eDirectory tree where ZfS servers reside.
WORKING_PATH	Working directory for the Server Policies and Server Software Packages components.

User-Defined Variables

User-defined variables are created in the Server Software Package component, Subscriber objects, and the Tiered Electronic Distribution policy. Policy packages do not recognize user-defined variables.

User-defined variables are not case sensitive.

Syntax:

%variable_name%

where *variable_name* is the name you give the variable when you define it. Spaces cannot be used in variable names. Use hyphens (-) or underscores (_) to separate words.

Variables defined in the Subscriber object are simple text substitutions. Text entered for the value of the variable will be substituted for the variable name.

Resolution of Variable Names

General variable definitions, such as those in the Tiered Electronic Distribution policy, provide default variable values for Subscribers where they have none defined. And, variables set in a Subscriber will override default variable values, such as those that were set in the Tiered Electronic Distribution policy.

However, for Server Software Packages, variable names are resolved differently:

1. Is the variable defined in the Server Software Package component? If so, use that value.
 - IMPORTANT:** A variable defined in a software package will override any value defined in the Subscriber.
2. Is the variable one of the predefined variables? If so, use that value.

3. Is the variable a Java environment variable? If so, use that value.

Nested Variables

Variables can be nested to any level. For example, you can do the following to automate destinations:

1. Define %Dest% as the destination volume and directory for a software package:
 - ♦ **Variable Name:** %Dest%
 - ♦ **Value:** %Vol%%Dir%
2. Define the %Vol% variable:
 - ♦ **Variable Name:** %Vol%
 - ♦ **Value:** %server_DN;attribute_name%
3. Define the %Dir% variable:
 - ♦ **Variable Name:** %Dir%
 - ♦ **Value:** \APPS (a directory on the volume)
4. On each server that will be processing software packages, locate the attribute_name defined in the value of %Vol% and enter the name as the volume where you want the software package extracted (such as DATA:).

The result is that when you create a software package, you can define the destination as simply %Dest%, which will resolve to the directory and volume specified at each target server. For example:

```
Server_001.Admin.Novell\DATA:\APPS
```

Creating a Variable

You can create variables in three locations:

- ♦ [“Creating Default Variables for All Subscribers” on page 582](#)
- ♦ [“Creating Variables for a Specific Subscriber” on page 583](#)
- ♦ [“Creating Variables for a Software Package” on page 583](#)

Creating Default Variables for All Subscribers

You can use the Tiered Electronic Distribution policy to create default variables for all Subscribers.

To create default variables:

- 1** In ConsoleOne[®], right-click a Service Location Package object > click Properties > click the check box for the Tiered Electronic Distribution policy to both select and enable it > click Properties > click the Variables tab.
- 2** Click Add.
- 3** Enter the name of the variable.

The name can be user-defined, an environment variable (Java or native), or a predefined variable.

- 4** Enter the value for the variable.
The value is what the variable will resolve to. It can also be another variable for nesting variables.
To ensure that extraction will take place, provide an absolute path to the Subscriber. For example, if the path is only the DATA volume, make sure the colon (:) is included, because it is a necessary part of the full path.
- 5** Enter a description (optional) > click OK.
- 6** Repeat **Step 2** through **Step 5** to create another variable for this Subscriber.
- 7** Click OK when you have finished creating the default variables > click OK to exit the policy package.

Creating Variables for a Specific Subscriber

To create variables for a specific Subscriber:

- 1** In ConsoleOne, right-click a Subscriber object > click the Variables tab.
- 2** Click Add.
- 3** Enter the name of the variable.
The name can be user-defined, an environment variable (Java or native), or a predefined variable.
- 4** Enter the value for the variable.
The value is what the variable will resolve to. It can also be another variable for nesting variables.
To ensure that extraction will take place, provide an absolute path to the Subscriber. For example, if the path is only the DATA volume, make sure the colon (:) is included, because it is a necessary part of the full path.
- 5** Enter a description (optional) > click OK.
- 6** Repeat **Step 2** through **Step 5** to create another variable for this Subscriber.
- 7** Click OK when you have finished creating variables for the Subscriber.

Creating Variables for a Software Package

To create a variables for a software package:

- 1** In ConsoleOne, right-click a software package > click the Variables tab.
- 2** Click Add.
New Variable #1 is defaulted in the Variables column.
- 3** To enter a different name for the variable, use the Backspace key to delete the default name > enter a new variable name > click the Tab key.
The name can be user-defined, an environment variable (Java or native), or a predefined variable.
- 4** Enter the value for the variable.
The value is what the variable will resolve to. It can also be another variable for nesting variables.

- 5 Repeat **Step 2** through **Step 4** to create another variable.
- 6 Click OK when you have finished creating variables for the software package.

Using a Variable to Change a Subscriber's Console Prompt

The Subscriber can use the value of the PROMPT variable as its server console prompt.

To set the PROMPT variable for a Subscriber's console prompt:

- 1 In ConsoleOne, right-click a Subscriber object > click Properties.
- 2 Click the Variables tab > click Add.
- 3 In the Variables dialog box, enter information for the following fields:
 - Variable:** Enter PROMPT as the variable name.
 - Value:** Enter the prompt text to be displayed. For example, %SERVER NAME% Subscriber > could display as:


```
Provo_01 Subscriber >
```
 - Description:** Enter a meaningful note (optional).
- 4 Click OK twice.

Using Variables to Control File Extraction

You can use variables to control the location that files are extracted to on the Subscriber. Any destination can be used as a variable defined in a Subscriber object by encapsulating it with the percent (%) symbol.

IMPORTANT: Any variable value specified in the Tiered Electronic Distribution policy is a default value and is overridden by variable values set in a Subscriber object.

For the location where files will be extracted, the destination root is identified in the File Grouping dialog box as a directory named \DESTROOT. This is the top-level directory used by a Subscriber to determine where to extract the file. The dialog box lets you build groups of directories under the DESTROOT directory.

The destination root can be specified as a known location (for example, %APP_DIR%). You can then go to the Variables tab on the Subscriber object and specify a value for this variable.

For example:

Variable	Value
APP_DIR	SYS:\APPS

To use a variable to set the location that files are extracted to:

- 1 In ConsoleOne, right-click the Subscriber object > click Properties.
- 2 Click the Variables tab > click Add.
- 3 Enter the name of the variable.
 - The name can be user-defined, an environment variable (Java or native), or a predefined variable.

- 4 Enter the value for the variable.

The value is what the variable will resolve to. It can also be another variable for nesting variables.

To ensure that extraction will take place, provide an absolute path to the Subscriber. For example, if the path is only the DATA volume, make sure the colon (:) is included, because it is a necessary part of the full path.

- 5 Enter a description (optional) > click OK > OK (to exit the properties).

- 6 Create a new Distribution object.

For information, see [“Creating a Distribution” on page 402](#).

- 7 In the Distribution object’s properties, click the Type tab > in the Select Type drop-down box, select File > click New Target.

- 8 Replace the default %DEST_VOLUME% with the variable name > click OK as necessary to exit the properties.

A directory named Dest_Volume is created by default in the Destination column. You should select this directory to change the destination root. To select it, click the actual directory name (DestRoot). You can then specify a known location or use a variable with surrounding percent symbols.

23 ZENworks Database

The following sections provide information for understanding and using the Novell® ZENworks® for Servers (ZfS) database in Policy and Distribution Services:

- ◆ “Understanding the ZENworks Database” on page 587
- ◆ “Determining How Many Databases You Need” on page 589
- ◆ “Installing, Setting Up, and Connecting To the ZENworks Database” on page 592
- ◆ “Creating a ZENworks Database Object” on page 596
- ◆ “Purging the Database” on page 597

Understanding the ZENworks Database

The following sections provide an understanding of the ZENworks database:

- ◆ “The Database File” on page 587
- ◆ “Database File Location” on page 587
- ◆ “The Database Object” on page 587
- ◆ “Running the Database” on page 289
- ◆ “Database Caching” on page 289
- ◆ “Database Information” on page 588
- ◆ “Coexisting Databases” on page 589

The Database File

Policy and Distribution Services uses a Sybase database file named ZFSLOG.DB. ZfS can function normally without the database, because it uses ZFSLOG.DB only to log information for Policy and Distribution Services reporting.

Database File Location

ZFSLOG.DB is normally located in the \ZENWORKS\PDS\DB directory on a server. Its location is determined when using the installation program. It can reside on both NetWare® and Windows servers.

The Database Object

A Novell eDirectory™ database object is created during installation. In its properties, you must list the location of the database file (ZFSLOG.DB), and you must configure the ZENworks Database

policy (Service Location Package) to specify the database object. The location and policy are necessary for the database file to be found for logging information.

Running the Database

On NetWare servers, the database is run by using the MGMTDBS.NCF file (located in the SYS:\SYSTEM directory), which is executed from AUTOEXEC.NCF.

On Windows servers, the database is run by using the Novell Sybase Database service.

Database Caching

Database files can become very large, which is why a 32 MB cache is recommended on the server where you are running the database. Caching will improve server performance because of how frequently information can be logged to ZFSLOG.DB.

Database Information

ZFSLOG.DB is used by Policy and Distribution Services to log successes and failures for the Server Policies or Tiered Electronic Distribution (TED) components. Policy information can be purged automatically according to a policy setting. TED information can be purged manually from the database object. For information on purging, see [“Purging the Database” on page 597](#).

ZFSLOG.DB does not contain any configuration information.

The Distributor is the only TED object that writes to the database.

The following information is written to ZFSLOG.DB by the agents:

Agent	Information
Policy/Package	Failed and successful policies Discovered and unenforceable policies Down Server policy status Server Software Packages and components
Distributor	Distribution status: <ul style="list-style-type: none">◆ When built, sent, and extracted◆ Successes (plus reasons) of builds and extractions◆ Failures (plus reasons) of a build, send, receive, and extraction Subscriber status Revision histories

For information on obtaining reports on the database information, see [Chapter 24, “Reporting,” on page 599](#).

The following provides information on gathering data for the database:

- ◆ A Distributor keeps track of each Subscriber in its routing hierarchy, so it knows which parent Subscribers have received a Distribution.

- ◆ The Distributor knows which Subscribers are at the end of a particular route, so it can know if Subscribers have not received a Distribution because a Subscriber higher up in the hierarchy failed to receive the Distribution.
- ◆ Subscribers send messages directly to the Distributor indicating that they have received a Distribution. The Distributor does not return a confirmation that it received the Subscriber's message.
- ◆ If a Distributor is not running when a "Successfully Received" message is sent from a Subscriber, this information will not be written to the database. Because a message receipt confirmation is not received by the Subscriber, it will not re send the message.

Coexisting Databases

You can have multiple ZENworks databases in the tree. The number you have depends on whether you want consolidated reporting and can live with the additional network traffic in a WAN environment.

If you do not require consolidated reports, you can install one database object for each of your WAN segments. This will eliminate writing to the database file over a WAN link by the Distributor.

For the server selected for a database file, you should not install a ZENworks for Desktops (ZfD) database object when a ZENworks database object exists for Policy and Distribution Services. The ZfD database object will replace the ZfS database object. However, you can install a ZfS ZENworks database object where a ZfD database object exists.

Server Inventory or Management and Monitoring Services database objects can be installed where a ZENworks database object exists (or the other way around) without any database object replacement problem.

Determining How Many Databases You Need

You can install the database to both NetWare and Windows servers.

The installation program checks the version of the Sybase engine before updating it. If it doesn't exist, or is an older version, Sybase software is installed.

IMPORTANT: Make sure you select a server for the database where you are installing the Subscriber/Policies option. The Purge Database option in the ZENworks for Servers policy (Distributed Server Package) works only if the Policy/Package Agent software and the ZFSLOG.DB file are located on the same server.

The installation program automatically creates a database object for each instance of the database that is installed. You can install only one instance of the database per run of the installation program. The database object will be installed to the same eDirectory container as the Server object for the server where the database file, ZFSLOG.DB, is also installed.

Review the following to understand whether to have multiple database files:

- ◆ [“Database Logging and TED Reporting” on page 589](#)
- ◆ [“Multiple Databases” on page 590](#)

Database Logging and TED Reporting

Policy and Distribution Services can function normally without using a ZENworks database, because it uses the ZFSLOG.DB file to only log information for reports. ZFSLOG.DB for Policy and Distribution Services does not contain any configuration information.

The Distributor Agent writes its distribution status information (built, received, extracted) to the database. The Policy/Package Agent writes policy enforcement information and Server Software Package installation information to the database. You will need a separate ZENworks Database policy for each of these agents, even though they might be writing to the same database file.

The ZENworks Database policy is associated (Service Location Package version) for the Distributor Agent. The policy is distributed (Distributed Server Package version) for the Policy/Package Agent.

ZFSLOG.DB contains information about Distributions (sent, received, extracted, and so on) and Policy Packages (enforced, failed, and so on). This information is used for the Policy and Distribution Services reports.

The Distributor is the only component that logs TED information to the database file (ZFSLOG.DB). The Policy/Package Agent logs reporting information to this file.

Policy and Distribution Services provides six predefined reports for the Server Policies component and four for the TED component. The report information is obtained from information logged to its database file. The following reports are available:

Server Policies Reports	TED Reports
Discovered Policies	Distribution Detail
Down Server Policy	Revision History
Packages	Revision History Failure
Failed Policies	Subscriber Detail
Successful Policies	
Unenforceable Policies	

A selected report displays all of the applicable Server Policies or TED information currently logged in the database. The criteria you can specify for a report include date ranges, specific Distributions, Distribution versions, and so on.

You might want multiple databases for specialized reporting. For more information, see [“Advantages” on page 591](#).

For information on reporting, see [Chapter 24, “Reporting,” on page 599](#).

Multiple Databases

Policy and Distribution Services supports multiple instances of the ZENworks database per tree. However, we recommend that you install only one instance of the database per tree. Review the following:

- ◆ [“Advantages” on page 591](#)
- ◆ [“Distributor Object Contexts and Multiple Databases” on page 591](#)
- ◆ [“Determining Whether You Need Multiple Databases” on page 592](#)

Advantages

The advantage in having only one database is that the Distribution information provided by all of the Distributor Agents and Policy/Package Agents can be displayed in a single report.

For example, with a single database, your software package information can be contained in one report:

- ◆ The Distributor Agent's information on building and sending the Software Package Distribution
- ◆ The Policy/Package Agent's information on extracting and installing the software package

The advantages in having multiple databases are:

- ◆ Minimizing traffic over slow WAN links

For example, having a separate database for Policy/Package Agent logging on its server's side of a WAN link.

- ◆ Providing individual databases for specialized reporting

For example, if you have one database for the Distributor Agent (distributions) and one for the Policy/Package Agent (policies), the build and send information for the Software Package and Policy Package types of Distributions will be written to the distributions instance of the database, and the software package installation and policy enforcement information will be written to the policies instance of the database.

Distributor Object Contexts and Multiple Databases

One ZFSLOG.DB file can receive log entries from multiple Distributors, and a Distributor can only log to one ZFSLOG.DB file. The following explains why:

- ◆ For a Distributor Agent to locate a database file, it must have a ZENworks Database policy (Service Location Package) associated with a context above the Distributor's object that points to the Database object, which contains the file's location in its properties. (Distributors receive their policies through association.)
- ◆ If you have separate databases installed on two or more of your Distributor servers, each database requires its own ZENworks Database policy for locating it (the policy points to the database's object, which contains its file's location).
- ◆ Only one Service Location Package (which contains the ZENworks Database policy) can be associated with a given context, such as the container holding your Distributor objects.
- ◆ Because only one Service Location Package can be associated with a given context, you must install your Distributor objects to different contexts to have multiple Distributors writing to their individual database files. Each Distributor would need its own database location policy that is associated with its own parent container.

For ease of management, you can keep your Distributor objects near each other by creating individual containers for each of them under the container where you would usually place all of them. Then you can associate the different Service Location Packages with their appropriate Distributor's unique parent containers.

- ◆ To have all of your Distributors write to the same database file, place each of their Distributor objects somewhere under the container where you associate the Service Location Package. They would all use the same database location policy.

Determining Whether You Need Multiple Databases

Consider the following to determine how many databases to have in the tree:

- ◆ **WAN Traffic:** TED does not perform a large number of database updates, so the actual impact on system resources should be minimal. The greatest impact could be the time it takes to perform the transaction. However, if you have slow WAN connections, you might not want database logging to occur over the WAN.
- ◆ **Multiple Distributors:** If you have multiple Distributors in the tree, you can have one database for each, or have them share one or more databases. The type of Distributor reporting you want should determine whether to have a separate database for each. For example, are your Distributors specialized in the types of Distributions they'll send?
- ◆ **Consolidated Reporting:** To have only one report for all of your TED information, install only one database object and file and have all TED Distributors log to that one file, regardless of WAN traffic considerations. Use the ZENworks Database policy (Service Location Package) to direct all Distributors to that database file.
- ◆ **Specialized Reporting:** You might want reports that are specific to a region or group of servers. You can install a database object and file for each such region and have the Distributors in those regions or server groups log to that database. Use a separate ZENworks Database policy (Service Location Package) to direct each Distributor to its desired database file.

Installing, Setting Up, and Connecting To the ZENworks Database

The ZENworks database should be installed on a server where policies are enforced. This is required so that you can use the ZENworks Database policy to locate ZFSLOG.DB.

The database object is automatically created in the tree when you run the installation program and select a server for the database.

The installation program can install only one database at a time. To install additional databases to the tree, you will need to perform the steps in the following sections for each database to be installed.

Perform the steps in the following sections to install and set up the database:

- ◆ [“Installing the Database” on page 592](#)
- ◆ [“Configuring the ZENworks Database Policy” on page 594](#)
- ◆ [“Connecting to the Database” on page 595](#)

Installing the Database

To install the ZENworks database:

- 1 On the workstation, insert the *ZENworks for Servers Program* CD or the ZENworks 6 Server Management Program CD.

The startup screen is displayed. If the startup screen is not automatically displayed after inserting the CD, you can start it by running WINSETUP.EXE at the root of the CD.

IMPORTANT: Installation from a remote CD is not supported unless there is a drive mapped on the workstation to that CD. For example, if you place the CD in a Windows NT server CD drive, then run the installation from a workstation, you must have a drive mapped to the CD drive of that NT server.

2 Click the ZENworks for Servers option > click the Install Policy and Distribution Services option.

3 Review the License Agreement > click Accept if you agree > click Next.

or

Click Decline > click Next to exit the installation program.

4 Browse and select the tree to install to (only one tree can be selected) > click Next.

The tree name is not case sensitive.

5 Click Next on the License page.

6 Click Add > browse for the server or multiple servers (use Shift or Ctrl) where you want to install the database.

7 Check the radio button under the Database column for each server where you want a database to be running.

You can select only one server per run of the installation program.

You might want a database for each Distributor to write its own information to. However, Distributors can share a database. Because the Distributor writes information to the database for all TED objects, the database should be installed on the same server as the Distributor to minimize network traffic.

If you have not previously installed the Zfs 2 database, enable this option for at least one Distributor. If you enable this option, the installation program checks all mounted volumes on the server to see if ZFSLOG.DB exists. If not, both the file and the database object will be installed. If the file exists, the database object will still be installed.

IMPORTANT: Make sure you select a server for the database where you are installing policies. The Purge Database option works only if the ZFS.NCF and ZFSLOG.DB files are on the same server.

8 To install the necessary software for the database objects, click the Copy Files and Create Objects for the Selected Components radio button.

If this option is not selected, software will not be installed and the database objects will not be created in the tree.

9 To have the installation program modify AUTOEXEC.NCF, check this box.

The Modify AUTOEXEC.NCF option will ensure that the database will be started.

10 To pause the installation and give you the opportunity to unload Java before continuing the installation, check the Pause Installation If JAVA.NLM Is Loaded On Target Server box.

If this box is not checked, the installation program skips any servers where Java is loaded and does not install the database. An error for each such server will be logged to:

```
C:\TEMP\_RESNumber.TXT
```

where *Number* is increased incrementally each time a new installation log is created.

11 When you have finished configuring the component options, click Next.

The Database page is displayed.

12 Select a volume for the database > click Next.

SYS: is not recommended because ZFSLOG.DB can become large. It is used for logging report information on server policies and TED usage.

The Summary page is displayed.

- 13** Review your selections > click Finish.

The installation program now copies files and installs the database objects.

WARNING: If you click Cancel, none of the work you did in the installation program is saved.

After the installation has finished, you can check the installation log file (see [Step 10](#)) to see if any components failed to install.

- 14** Continue with setting up the database (see [“Configuring the ZENworks Database Policy” on page 594](#)).

Configuring the ZENworks Database Policy

You must set up a database locator policy so that information can be logged to the database.

The Distributor Agent requires a database policy that is associated. The Policy/Package Agent requires a database policy that is distributed to each Subscriber server where the agent is installed.

The Distributor Agent writes distribution information, and the Policy/Package Agent writes policy information.

Perform the following applicable tasks:

- ◆ [“For the Distributor Agent” on page 594](#)
- ◆ [“For the Policy/Package Agent” on page 595](#)

For the Distributor Agent

To configure the required attributes for the ZENworks Database policy:

- 1** In ConsoleOne[®], browse eDirectory for the container you created specifically for Policy Package objects.

If necessary, create the container object.

- 2** Right-click the policies container > click New > click Policy Package to open the Policy Package Wizard.

- 3** Under Policy Packages, select Service Location Package > click Next.

- 4** Name the package > click Next > click Finish to create the package.

Name the package so that it is identified with its ZENworks Database object.

- 5** Right-click the Service Location Package > click Properties > click the Policies tab.

If the box under the Enabled column is not checked for the ZENworks Database policy, click it before clicking Properties. A policy must be enabled to activate the Properties button.

- 6** Click the check box under the Enabled column for the ZENworks Database policy to enable it > click Properties.

- 7** Click to make sure you are viewing the Policy/Distribution Management tab.

- 8** Browse for a Database object (or enter its DN) > click Apply > click OK.

For example, the Database object might read:

```
zfs Database.Development.Novell
```

The Database object was automatically created when you installed the Database. It is located in the same container as the Server object where the database was installed.

- 9** Click the Associations tab > click Add.
- 10** Browse to the container containing the Distributor objects > click Apply > click OK.
If you have your Distributor objects in different containers, add the other containers to the list.
The Service Location Package object must be associated so that ZFSLOG.DB can be found by the Distributor Agent for logging information.
- 11** Click OK when finished associating the Service Location Package.

For the Policy/Package Agent

To configure the required attributes for the ZENworks Database policy:

- 1** In ConsoleOne, browse eDirectory for the container you created specifically for Policy Package objects.
If necessary, create the container object.
- 2** Right-click the policies container > click New > click Policy Package to open the Policy Package Wizard.
- 3** Under Policy Packages, select Distributed Server Package > click Next.
- 4** Name the package > click Next > click Finish to create the package.
Name the package so that it is identified with its ZENworks Database object.
- 5** Right-click the Distributed Server Package > click Properties > click the Policies tab.
- 6** Click the check box under the Enabled column for the ZENworks Database policy to enable it > click Properties.
- 7** Click to make sure you are viewing the Policy/Distribution Management tab.
- 8** Browse for a Database object (or enter its DN) > click Apply > click OK.
For example, the Database object might read:

`ZFS Database.Development.Novell`
The Database object was automatically created when you installed the Database. It is located in the same container as the Server object where the database was installed.
- 9** Click OK when finished.
- 10** Distribute the Distributed Server Package object.
For information on distributing policy packages, see [“Distributing Policies” on page 487](#).
The Distributed Server Package object must be distributed so that ZFSLOG.DB can be found by the Policy/Package Agent for logging information.

Connecting to the Database

To make sure that the database will be written to by the Policy/Package Agent and the Distributor Agent:

- 1** On a server, load the Policy/Package Agent by doing the following:

Server Platform	Agent Startup Method
Windows	<ol style="list-style-type: none"> 1. Open the Control Panel. 2. Click Services (in Windows 2000, Services is under Admin Tools). 3. Click Novell ZfS Policies > click Start.
NetWare	<code>SYS:\ZENWORKS\PDS\SMANAGER\ZFS.NCF</code>
Solaris or Linux	<code>/usr/ZENworks/pds/smanager/ZFSSRV.sh</code>

Note whether a message is displayed indicating that the Policy/Package Agent has connected to the database.

- 2** On a server, load the Distributor Agent by doing the following:

Server Platform	Agent Startup Method
Windows	<ol style="list-style-type: none"> 1. Open the Control Panel. 2. Click Services (in Windows 2000, Services is under Admin Tools). 3. Click Novell ZfS Distribution > click Start.
NetWare	<code>SYS:\ZENWORKS\PDS\SMANAGER\TED.NCF</code>
Solaris or Linux	<code>/usr/ZENworks/pds/smanager/TEDSRV.sh</code>

Note whether a message is displayed indicating that the Distributor Agent has connected to the database.

- 3** Repeat **Step 1** and **Step 2** for each server where a Policy/Package Agent or Distributor Agent has been installed.

IMPORTANT: You must repeat **Step 2** for each Distributor server because the Distributor Agent must be started or restarted to connect with the database.

- 4** To determine whether the Policy/Package Agent or the Distributor Agent is writing to the database, do the following for each agent:

- 4a** At a NetWare server's console prompt, view the monitor while the agent is loading.

A message should display that states whether the agent connected with the database.

- 4b** If the message indicates that the agent did not connect to the database, you should check the following:

- ◆ Is the database is running on the server?
- ◆ Is there a database object that has its Policy/Distribution Management tab set up with the server where the database file is installed?
- ◆ Is there an effective ZENworks Database policy pointing to the database object?

Creating a ZENworks Database Object

The ZENworks Database object might not exist for the following reasons:

- ◆ You have inadvertently deleted the object

- ♦ You did not select to install the database when you installed Policy and Distribution Services

If the database object does not exist in the tree, you can manually create it.

To create a database object:

- 1** In ConsoleOne, right-click a location in the tree for the database object > click New > Object > ZENworks Database.
- 2** Enter a database name.
- 3** Click the Define Additional Properties check box > click OK.
- 4** On the ZENworks Database tab, click either the Server DN or Server IP Address radio button.
One of these location IDs could already be the default. If not, enter the information, which should be for the server where ZFSLOG.DB resides.
- 5** Click the eDirectory Rights tab > Trustees of This Object > Add Trustee > select [Public].
The database object must be assigned a trustee of Public or the Policy/Package Agent will display messages that it cannot connect with the database nor read the ZENworks for Servers policy.
- 6** Click OK.
If you click Cancel, none of the information you added or changed on any of the tabs will be saved. However, the database object will remain on the tree.
- 7** Set up the ZENworks Database policy.
For steps to specify the location of a database, see [“ZENworks Database” on page 484](#).
- 8** Associate the Service Location Package with a container above where the Distributor object resides.

Purging the Database

Because Policy and Distribution Services logs all successes and failures for the Server Policies or TED components, ZFSLOG.DB can quickly grow in size. Therefore, you should periodically purge ZFSLOG.DB.

Purging of policy information is done automatically according to the schedule you set whenever ZFS.NCF is started on a server where ZFSLOG.DB resides.

You can manually purge a selected database of all TED information older than a specific date and time.

To manually purge a database:

- 1** In ConsoleOne, right-click the database object > click Purge.
- 2** In the Purge Database dialog box, select a date and time > click OK.

24 Reporting

Novell® ZENworks® for Servers (ZfS) provides predefined reports for the Policy and Distribution Services components. There are six reports for Server Policies, and four for Tiered Electronic Distribution (TED).

Policy and Distribution Services reports are accessed from the menu options of certain ZfS objects. All reports can be accessed from the ZENworks Database object, and the TED reports can be accessed from the Subscriber and Distribution objects. These reports should not be accessed from the ConsoleOne® reporting feature.

A selected report displays all of the applicable Server Policies or TED information currently logged in the database. There are options for defining the parameters of some reports, such as date ranges, or selecting Policy Package objects or TED objects.

A ZENworks database file (ZFSLOG.DB) is used to store the report information. Once you have installed and run the database and data has been placed in ZFSLOG.DB, Policy and Distribution Services reporting is enabled.

You can create custom reports using the table definitions listed under [“Creating Customized Reports” on page 605](#).

Review the following:

- ◆ [“Storing Report Information” on page 599](#)
- ◆ [“Reporting Scope for TED Objects” on page 600](#)
- ◆ [“Reporting on the Successes and Failures of Distributions” on page 600](#)
- ◆ [“Generating Reports” on page 600](#)
- ◆ [“Report Descriptions” on page 601](#)
- ◆ [“Creating Customized Reports” on page 605](#)

Storing Report Information

ZFSLOG.DB will not receive information for reporting unless the following actions have taken place:

- ◆ The ZENworks Database policy (Service Location Package) has been configured and enabled
- ◆ The Policy/Package Agent has been either refreshed from the server console or ZfS has been restarted
- ◆ The Distributor Agent has been restarted (not refreshed) after the ZENworks Database policy has been enabled

Reporting Scope for TED Objects

The Distributor object is the only TED object that writes to the ZENworks database file. Each Distributor object normally has its own ZENworks Database object and database file (ZFSLOG.DB). Therefore, report information is given only for the particular Distributor object selected for a report.

Reporting on the Successes and Failures of Distributions

Reporting gives a high-level overview of which nodes succeeded. All known error conditions are caught and error conditions are reported to the database. However, when a process status is in progress, errors can occur or failures can occur on the node that are not caught (for example, the machine went down or the process was killed).

The Distribution-level reports show the view from the Distributor side and are very useful for checking which Subscribers succeeded or failed to receive a particular Distribution. The Subscriber reports are used to determine which Distributions a single Subscriber has received.

Subscribers that did not attempt to receive the Distribution (because they were not set up correctly or were not running) will not have information displayed on the report. You can compare the number expected against the actual numbers and look for missing Subscribers on the report. Once Subscribers are set up and have been functioning, this should not be a common problem.

Generating Reports

To generate a Policy and Distribution Services report:

- 1 In ConsoleOne, right-click one of the following:

- Distribution object
- Subscriber object
- ZENworks Database object

The ZENworks Database object must be one that has its Policy/Distribution Management tab configured (not the Inventory Management tab).

- 2 Click Reports.
- 3 Select a report.

If you clicked a Distribution object in **Step 1**, you can select from the following reports:

- Distribution Detail
- Revision History
- Revision History Failure
- Subscriber Detail

If you clicked a Subscriber object in **Step 1**, you can select the Distribution Detail report.

If you clicked the database object in **Step 1**, you can select from the following server policy reports (as well as the above Distribution reports):

Discovered Policies	Server Down Process Policy
Failed Policies	Successful Policies
Packages	Unenforceable Policies

4 Select the reporting criteria.

If you need more detail on reporting criteria or content, see [“Report Descriptions” on page 601](#).

5 Click Run Selected Report.

The View Report dialog box is used to display the generated report.

6 To print the report, click File > Print.

or

To export the report, click File > Export Report.

Report Descriptions

The following sections describe the Policy and Distribution Services reports:

- ◆ [“TED Reports” on page 601](#)
- ◆ [“Server Policy Reports” on page 603](#)

TED Reports

There are four predefined TED reports:

- ◆ [“Distribution Detail Report” on page 601](#)
- ◆ [“Revision History Report” on page 602](#)
- ◆ [“Revision History Failure Report” on page 602](#)
- ◆ [“Subscriber Detail Report” on page 602](#)

Distribution Detail Report

Displays a detailed, time-line style history of Distributions for the selected Subscribers (for more information, see the [Subscriber](#) bullet), including:

- ◆ Distributions Sent
- ◆ Distributions Received
- ◆ Distributions Extracted (including start time, end time, and completion code)

Sorting is by time; grouping is by Distribution name and version.

The report criteria include:

- ◆ **Subscriber:** If you right-clicked a Subscriber object, it appears in the Subscriber field and the report only displays information for the receive and extract actions performed by this Subscriber. Information for parent Subscribers will also display a Received Stage heading.
If you right-clicked the database or Distribution object, the report includes all actions that have occurred with a Distribution. In other words, information for all Subscribers involved is displayed.
- ◆ **Latest Version Only:** Uncheck to include versions that are within the specified date range.
- ◆ **Select the Date Range Criteria for the Report:** Specify the range.

Revision History Report

Displays a history of a Distribution package's versions, including:

- ◆ Distribution (DN of package)
- ◆ Distributor (DN of object)
- ◆ Version Number
- ◆ Creation Date/Time
- ◆ Distribution Size

Sorting is by version number.

The report criteria include:

- ◆ **Distribution:** If you right-clicked a Distribution object, it appears in the Distribution field. If you right-clicked the database object, you will need to browse for the Distribution object.

Revision History Failure Report

Displays the versions of the Distribution that failed during creation, including:

- ◆ Distribution (DN of package)
- ◆ Distributor (DN of object)
- ◆ Creation Date and Time
- ◆ Error Description

Sorting is by version.

The report criteria include:

- ◆ **Distribution:** If you right-clicked a Distribution object, it appears in the Distribution field. If you right-clicked the database object, you will need to browse for the Distribution object.

Subscriber Detail Report

Displays status information for the Subscribers that received the Distribution, including:

- ◆ Distribution and Version
- ◆ Subscriber (DN of object) and Subscriber's Address
- ◆ Channel Name
- ◆ Source (DN of Distributor)
- ◆ Stage
- ◆ Status
- ◆ Date and Time
- ◆ Error Description

Sorting is by Subscriber/Parent Subscriber, then Stage.

The report criteria include:

- ◆ **Distribution:** If you right-clicked a Distribution object, it appears in the Distribution field. If you right-clicked the database object, you will need to browse for the Distribution object.

- ◆ **Version Number:** If Distribution versions exist, you can choose one from the drop-down menu. Select All to include all versions.
- ◆ **Distribution Stage:** You can select All, Extract, or Receive.
- ◆ **Distribution Status:** You can select All, Success, or Not Success.

Server Policy Reports

Note that for all server policy reports, the default date ranges are for the current date (from midnight to midnight).

There are six predefined server policy reports:

- ◆ “Discovered Policies Report” on page 603
- ◆ “Server Down Process Report” on page 603
- ◆ “Failed Policies Report” on page 604
- ◆ “Packages Report” on page 604
- ◆ “Successful Policies Report” on page 604
- ◆ “Unenforceable Policies Report” on page 605

Discovered Policies Report

Displays the servers that have discovered policies within the specified packages, including:

- ◆ Package (DN)
- ◆ Server DN
- ◆ Server Name
- ◆ OS Name and OS Version
- ◆ Date/Time of Discovery

Sorting is by package, then by context/server name, maintaining the tree’s hierarchy. For example, MYSERVER.PRIV.NOVELL is sorted NOVELL, PRIV, MYSERVER.

The report criteria include:

- ◆ **Package:** Select a policy package from the drop-down list or select All.
- ◆ **Policy Type:** You can select All, Server Down Process, Scheduled Down, SNMP Trap Targets, Community Strings, Set Parameters, Script, Text File, Scheduled Load/Unload, or Database Location.
- ◆ **Select the Date Range Criteria for the Report:** Specify the range.

Server Down Process Report

For a selected server or all servers in the tree, displays Server Down Process policy information, including:

- ◆ Down Action and Code for each policy

Sorting is by server name only.

The report criteria include:

- ◆ **Server:** Select a server from the drop-down list or select All.
- ◆ **Select the Date Range Criteria for the Report:** Specify the range.

Failed Policies Report

For all servers in the tree, displays all policies that have failed, including:

- ◆ Package (DN)
- ◆ Server DN
- ◆ Server Name
- ◆ OS Name
- ◆ Date/Time of Failure
- ◆ Reason for Failure (Description)

Sorting is by context/server name, maintaining the tree's hierarchy. For example, MYSERVER.PR.V.NOVELL is sorted NOVELL, PRV, MYSERVER.

The report criteria include:

- ◆ **Package:** Select a policy package from the drop-down list or select All.
- ◆ **Failure Type:** You can select All, Failed, Unenforceable, or Partial Enforcement.
- ◆ **Policy Type:** You can select All, Server Down Process, Scheduled Down, SNMP Trap Targets, Community Strings, Set Parameters, Script, Text File, Scheduled Load/Unload, or Database Location.
- ◆ **Select the Date Range Criteria for the Report:** Specify the range.

Packages Report

Displays information on Server Software Packages and their components, including:

- ◆ Success status of each package
- ◆ Success status of each component

Sorting is by context/server name, maintaining the tree's hierarchy. For example, MYSERVER.PR.V.NOVELL is sorted NOVELL, PRV, MYSERVER.

The report criteria include:

- ◆ **Package:** Select a software package from the drop-down list or select All.
- ◆ **Server:** Select a server from the drop-down list or select All.
- ◆ **Select the Date Range Criteria for the Report:** Specify the range.

Successful Policies Report

For all servers in the tree, displays all policies that have been successfully enforced, including:

- ◆ Package (DN)
- ◆ Server DN
- ◆ Server Name
- ◆ OS Name

- ◆ Date/Time of Run
- ◆ Action Code

Sorting is by context/server name, maintaining the tree's hierarchy. For example, MYSERVER.PR.V.NOVELL is sorted NOVELL, PRV, MYSERVER.

The report criteria include:

- ◆ **Package:** You can specify a single policy package or select All.
- ◆ **Success Type:** You can select All, Change, or No Change.
- ◆ **Policy Type:** You can select All, Server Down Process, Scheduled Down, SNMP Trap Targets, Community Strings, Set Parameters, Script, Text File, Scheduled Load/Unload, or Database Location.
- ◆ **Select the Date Range Criteria for the Report, From/To:** Specify the range.

Unenforceable Policies Report

Displays all unenforceable policies because of the absence of an enforcer on a server for all servers in the tree, including:

- ◆ Package (DN)
- ◆ Server DN
- ◆ Server Name
- ◆ OS Name and OS Version

Sorting is by package, then by server name.

The report criteria include:

- ◆ **Package:** Select a policy package from the drop-down list or select All.
- ◆ **Select the Date Range Criteria for the Report:** Specify the range.

Creating Customized Reports

Using the following database information you can create custom reports for the Server Policies and TED components.

However, for TED objects such as a Subscriber or the External Subscriber, you should use ZENworks reporting options (see [Chapter 24, “Reporting,” on page 599](#)) or iManager ([Chapter 15, “Novell iManager,” on page 359](#)) for determining the status of Distributions or policies.

The database file (ZFSLOG.DB) contains the following information:

- ◆ [“Default Sybase Database User ID and Password” on page 605](#)
- ◆ [“Server Policies Database Contents” on page 606](#)
- ◆ [“TED Database Contents” on page 612](#)

Default Sybase Database User ID and Password

The Sybase database (ZFSLOG.DB) that ships with ZfS has the following default user ID and password:

User ID: dba

Password: sql

Server Policies Database Contents

Following are the database table definitions for server policies.

SERVERS

Contains one record for each server running the Policy/Package Agent.

Field Name	Type		Use
SERVERID	integer	not null	Unique number that is automatically assigned.
SERVERNAME	varchar	not null	The short name of the server as seen on the console prompt.
SERVERDN	varchar		DN of the Server object in eDirectory (dot separated).
REVERSEDN	varchar	not null	SERVERDN in reverse order and backslash (\) delimited.
OSNAME	varchar		Name of the operating system, such as NetWare 5.1.
OSVERSION	char		Version of the operating system, such as 5.1, 6.0, and so on.
TREENAME	varchar		Name of the eDirectory tree containing the server.

Primary key (SERVERID)

SERVERIP

Contains one record for each server running the Policy/Package Agent.

Field Name	Type		Use
SERVERIPKEY	integer	not null	Assigned automatically: Default Auto increment.
SERVERID	integer	not null	Links to the SERVERS table.
IPADDRESS	varchar	not null	Server's IP address.

Primary key (SERVERID) REFERENCES SERVERS

Primary key (SERVERIPKEY)

PACKAGES

Contains one record for each version of a software package that the Policy/Package Agent has attempted to process.

Field Name	Type		Use
PACKAGEGUID	char	not null	Assigned automatically: Assigned Automatically.
PACKAGENAME	char		Name of .CPK file or policy package.
PACKAGEDESC	char		Description contained in a Server Software Package component.
PACKAGEVERSION	char		Version of the software package.
BUILDDATE	integer		Date the software package was compiled.

Primary key (PACKAGEGUID)

POLICIES

Contains one record for each policy or policy package combination.

Field Name	Type		Use
POLICYID	integer	not null	A globally unique ID.
POLICYDN	varchar		The DN of the eDirectory policy object.
POLICYPACKAGE	varchar		The DN of the policy package the policy belongs to.
POLICYCLASS	varchar		The class or type of policy. For definitions, see "Valid Entries for POLICYCLASS" on page 607 .
POLICYTREENAME	varchar		The name of the tree the policy object is in.

Primary key (POLICYID)

Valid Entries for POLICYCLASS

zenZFSServerDowningPolicy
zenZFSScheduleDownPolicy
zenZFSSetServerParamPolicy
zenZFSServerScriptPolicy
zenZFSTextFilePolicy
zenZFSScheduledRunPolicy
zenZFSZFSPolicy
zenZFSCommunityPolicy
zenZFSSNMPTrapTargetPolicy
zenZFSSMTPHostPolicy
zenZFSDatabaseLocationPolicy
zenZFSLicenseLocationPolicy
zenZFSTEDPolicy

POLICYACTION

Contains one record for each action performed.

Field Name	Type		Use
POLICYACTIONKEY	integer	not null	Assigned automatically: Default Auto increment.
POLICYID	integer	not null	Links to the POLICIES table.
SERVERID	integer	not null	Links to the SERVERS table.
CREATIONDATE	timestamp		Time stamp of the action.
DESCRIPTION	varchar		Undefined string describing an error.
CODE	integer		Code representing the result of the action. For definitions, see "Valid Entries for CODE" on page 608.
ACTIONCODE	integer		The action being performed. For definitions, see "Valid Entries for ACTIONCODE" on page 608.

Primary key (POLICYACTIONKEY)

Valid Entries for CODE

RC_POL_SUCCESS	= 0
RC_POL_PARTIAL_SUCCESS	= 1
RC_POL_FAILURE	= -1
RC_POL_EMPTY	= -2

Exception: If the value in the ACTIONCODE field is AC_POL_DOWN_CONNECTIONS or AC_POL_DOWN_DISCONNECTIONS, then the value of CODE is either the current number of active connections, or the number of forced disconnects.

Note that a number 1 in the CODE field can mean one of the following:

- ◆ There was a partial success
- ◆ There is one active connection
- ◆ There was one forced disconnect

This is because the meaning of the entry in the CODE field is determined by the content of the ACTION CODE field.

Valid Entries for ACTIONCODE

AC_POL_DISCOVERED	= 101
AC_POL_SCHEDULED	= 102
AC_POL_APPLIED	= 103
AC_POL_APPLIED_CHANGE	= 104
AC_POL_NO_ENFORCER	= 105
AC_POL_DOWN_CONNECTIONS	= 106
AC_POL_DOWN_DISCONNECTIONS	= 107

AC_POL_DOWN_UNLOAD	= 108
AC_POL_DOWN_EMAIL	= 109
AC_POL_DOWN_NOTIFY	= 110
AC_POL_DOWN_CANCELED	= 111
AC_POL_DOWN_IGNORED	= 112
AC_POL_DOWN_REQUESTED	= 113

PACKAGEACTION

Contains one record for each action taken on a Server Software Package.

Field Name	Type	Use
PACKAGEACTIONID	integer	not null Assigned automatically: Default Auto increment.
PACKAGEGUID	char	not null Links to the PACKAGES table.
SERVERID	integer	not null Links to the SERVERS table.
CREATIONDATE	timestamp	Time stamp of the action.
DESCRIPTION	varchar	For definitions, see “Valid Entries for DESCRIPTION” on page 609.
CODE	integer	Code representing the results of the action. For definitions, see “Valid Entries for CODE” on page 609.
ACTIONCODE	integer	Code representing the action being performed. For definitions, see “Valid Entries for ACTIONCODE” on page 610.
STARTEDPACKAGEACTIONID	integer	0 = started running the package, or when the new action is logged then the PACKAGEACTIONID of the new action replaces the 0.

Primary key (PACKAGEACTIONID)

Valid Entries for DESCRIPTION

Started package
Finished rollback
Error description
Or it is empty

Valid Entries for CODE

Success = 0
Failure = 1
Partial = 2

Valid Entries for ACTIONCODE

AC_PACKAGE_INSTALL	= 0
AC_PACKAGE_ROLLBACK	= 1
AC_PACKAGE_INSTALL_STARTED	= 2
AC_PACKAGE_ROLLBACK_STARTED	= 3

SOFTWARECOMPONENTACTION

Contains one record for each server Server Software Package component.

Field Name	Type	Use
SOFTWARECOMPONENTACTIONKEY	integer	not null Assigned automatically: Default Auto increment.
PACKAGEACTIONID	integer	not null Links to the PACKAGEACTION table.
NAME	char	not null Name of the software component.
CREATIONDATE	timestamp	Time stamp of the action.
DESCRIPTION	varchar	The first record for the component the description is the description entered by the user when the component was created. As the components finish the description is one of those defined under "Valid Entries for DESCRIPTION" on page 610.
CODE	integer	Code representing the results of the action. For definitions, see "Valid Entries for CODE" on page 611.
ACTIONCODE	integer	Code representing the action being performed. For definitions, see "Valid Entries for ACTIONCODE" on page 611.

Primary key (SOFTWARECOMPONENTACTIONKEY)

Valid Entries for DESCRIPTION

- Did not meet requirements
- Error processing requirements
- Pre-install load/unload
- Error pre-install load/unload
- Pre-install scripts
- Error pre-install scripts
- Copy file changes

Error processing copy file
Text file changes
Error processing text files
NetWare SET parameters
Error processing NetWare SET parameters
Registry process
Error processing Registry
NetWare products process
Error in NetWare products process
Post-install script process
Error in post-install script process
Post-install load/unload process
Error in post-install load/unload process

Valid Entries for CODE

Success = 0
Failure = 1
Partial = 2

Valid Entries for ACTIONCODE

Started = 200
Pre-Load = 201
Pre-Scripts = 202
Copy File Changes = 203
Text File Changes = 204
Set Parameters = 205
Registry = 206
Products.dat = 207
Post Scripts = 208
Post Load = 209
Requirements = 210

Foreign Keys

Foreign keys set up relationships between tables.

POLICYACTION

"add foreign key (POLICYID) references POLICIES (POLICYID)"

POLICYACTION

"add foreign key (SERVERID) references SERVERS (SERVERID)"

PACKAGEACTION

"add foreign key (PACKAGEGUID) references PACKAGES (PACKAGEGUID)"

PACKAGEACTION

"add foreign key (SERVERID) references SERVERS (SERVERID)"

SOFTWARECOMPONENTACTION

"add foreign key (PACKAGEACTIONID) references PACKAGEACTION (PACKAGEACTIONID)"

TED Database Contents

Following are the database table definitions for TED.

- ◆ [“TAB_NODE” on page 612](#)
- ◆ [“TAB_CHANNEL” on page 612](#)
- ◆ [“TAB_DISTRIBUTION” on page 613](#)
- ◆ [“TAB_DIST_VERSION” on page 613](#)
- ◆ [“TAB_DIST_ACTION” on page 614](#)
- ◆ [“TAB_CHANNEL_DISTRIBUTION” on page 615](#)
- ◆ [“Foreign Keys” on page 615](#)

TAB_NODE

Contains one record for each Distributor, Subscriber, and External Subscriber in the tree.

Field Name	Type	Use
ID	numeric(8,0) identity not null	Unique number automatically assigned.
NAME	varchar(255)	TED object DN.
TYPE	char	"D"=Distributor "T"=Subscriber (Transceiver)
NETWORK_ADDRESS	varchar(255)	IP address of server.
SERVER_NAME	varchar(255)	Not currently used.

Primary key (ID)

Unique (NAME)

TAB_CHANNEL

Contains one record for each Channel object in the tree.

Field Name	Type			Use
ID	numeric(8,0)	identity	not null	Unique number automatically assigned.
NAME	varchar(255)		not null	DN of Channel object.

Primary key (ID)
Unique (NAME)

TAB_DISTRIBUTION

Contains one record for each Distribution object in eDirectory.

Field Name	Type			Use
ID	numeric(8,0)	identity	not null	Unique number automatically assigned.
NAME	varchar(255)		not null	DN of Distribution object.
DISTRIBUTOR_ID	numeric(8,0)		not null	Links to the TAB_NODE table.

Primary key (ID)
Unique (NAME)

TAB_DIST_VERSION

Contains one record for each version of a Distribution and it is linked to the TAB_DISTRIBUTION table.

Field Name	Type			Use
ID	numeric(10,0)	identity	not null	Unique number automatically assigned.
DISTRIBUTION_ID	numeric(8,0)		not null	Links to the TAB_DISTRIBUTION table.
VERSION	bigint		not null	Time stamp of the version.
SIZE	integer		not null	Size of DISTFILE.TED (the file containing the Distribution).
TIMESTAMP	datetime		not null	Time stamp when the entry was made to the database.
DIRECT_ROUTING	bit		not null	Not used at the current time.
LATEST_VERSION	bit		not null	Latest version of this Distribution. Used internally to keep track of the latest version.

Primary key (ID)
Unique (DISTRIBUTION_ID, VERSION)

TAB_DIST_ACTION

Contains multiple records for each Distribution version for Send, Received, and Extracted.

Field Name	Type	identity	not null	Use
ID	numeric(12,0)			Unique number automatically assigned.
DIST_VERSION_ID	numeric(10,0)			Links to the TAB_DIST_VERSION table.
NODE_ID	numeric(8,0)			Links to the TAB_NODE table for the node performing the following tasks: Create Send Receive Extract Post process
TIMESTAMP	datetime			Time stamp when the action was logged into the database.
STAGE	char			"C"=Create "S"=Send "R"=Receive "E"=Extract "P"=Post process
STATUS	char			"S"=Success "F"=Failure "P"=In process
STATUS_TIMESTAMP	datetime			Time stamp when the record was updated.
REASON_TEXT	varchar(255)			Reason for success or failure. For definitions, see "Valid Entries for REASON_TEXT" on page 614.
CHANEL_DIST_ID	numeric(8,0)			Links to the TAB_CHANNEL_DISTRIBUTION table.

Primary key (ID)

Valid Entries for REASON_TEXT

The following are valid entries for the REASON_TEXT field name:

- ◆ "The Distribution was not received because this Subscriber does not meet the platform restrictions."

Self explanatory.

- ◆ "The Distribution was shut down before it was received."

This one is received in one of two situations: 1) we get a new configuration on the Subscriber so it needs to be updated before it can receive the Distribution; or, 2) we have a signature

exception, such as the Subscriber cannot trust the Distribution came from a Distributor it trusts.

- ◆ "The Distribution was terminated before it was received."

The Distribution was cancelled for a controlled reason.

- ◆ "There was an error receiving the Distribution."

This is a failure because something unexpected failed. For example, a socket exception, transport exception, and so on.

TAB_CHANNEL_DISTRIBUTION

Contains one record for each Channel/Distribution.

Field Name	Type	Use
ID	numeric(8,0) identity not null	Unique number automatically assigned
CHANNEL_ID	numeric(8,0) not null	Links to the TAB_CHANNEL table.
DISTRIBUTION_ID	numeric(8,0) not null	Links to the TAB_DISTRIBUTION table.
TIMESTAMP	datetime not null	Time stamp for when the Distribution was built.

Primary key (ID)

Unique (CHANNEL_ID, DISTRIBUTION_ID)

Foreign Keys

Foreign keys set up relationships between tables.

TAB_DISTRIBUTION

```
" add foreign key FK_TAB_DIST_REF_591_TAB_NODE (DISTRIBUTOR_ID)" + " references  
TAB_NODE (ID) on update restrict on delete restrict;"
```

TAB_DIST_VERSION

```
" add foreign key FK_TAB_DIST_REF_37_TAB_NODE (DISTRIBUTOR_ID)" + " references  
TAB_DISTRIBUTION (ID) on update restrict on delete restrict;"
```

TAB_DIST_ACTION

```
" add foreign key FK_TAB_DIST_REF_380_TAB_NODE (DIST_VERSION_ID)" + "  
references TAB_DIST_VERSION (ID) on update restrict on delete restrict;"
```

TAB_DIST_ACTION

```
" add foreign key FK_TAB_DIST_REF_1525_TAB_NODE (NODE_ID)" + " references  
TAB_NODE (ID) on update restrict on delete restrict;"
```

TAB_CHANNEL_DISTRIBUTION

" add foreign key FK_TAB_DIST_REF_572_TAB_DIST (DISTRIBUTION_ID)" + " references
TAB_DISTRIBUTION (ID) on update restrict on delete restrict;";

TAB_CHANNEL_DISTRIBUTION

" add foreign key FK_TAB_DIST_REF_572_TAB_CHAN (CHANNEL_ID)" + " references
TAB_CHANNEL (ID) on update restrict on delete restrict;";

B

Server Console Commands

This section is referenced from other sections.

You can perform some of the Novell® ZENworks® for Servers (ZfS) functions using command line entries on a NetWare® server console. The server commands documented here are those that are applicable to ZfS Server Policies and Tiered Electronic Distribution (TED).

For ways to perform the server console commands in a Web browser using the ZfS Management role in Novell iManager, see [Chapter 15, “Novell iManager,” on page 359](#).

A ZfS console command that is typed on a server console is executed only on that server. For more information, review the following sections:

- ♦ [“ZfS Console Commands” on page 617](#)
- ♦ [“Java Console Commands” on page 620](#)

ZfS Console Commands

The following table lists the ZfS server console commands with short descriptions of the commands. The table also indicates at which server console prompt a command can be given.

The column heading M is for the server’s main console prompt, Z for the ZfS prompt, and T for the TED prompt. Under a console prompt column, a Y indicates that the command can be issued at that prompt and a – indicates that the command cannot be issued at that prompt.

Command	M	Z	T	Description
HELP	Y	Y	Y	Displays a list of available commands. Only the commands applicable to a component will be displayed.
HELP <i>command</i>	Y	Y	Y	Displays help for the specified command.
CLS	Y	Y	Y	Clears the screen. Useful for quickly recognizing which information is new when you type a command.

Command	M	Z	T	Description
DOWN <i>option</i>	Y	Y	-	<p>This is similar to the command used on the server's main console prompt. However, if you use DOWN at the ZfS prompt, server policy settings for downing the server will be followed.</p> <p>For the ZfS prompt, this command has several options:</p> <ul style="list-style-type: none"> ♦ DOWN SERVER: Downs the server only; does not bring it back up. ♦ DOWN STATUS: Displays the current down status. ♦ DOWN RESTART: Downs the server, then restarts it. ♦ DOWN RESET: Downs the server, then resets it. ♦ DOWN CANCEL: Allows you to cancel the down, up to when the server is actually taken down. This will not leave the server in an unusable state. ♦ DOWN !: Causes the down process to execute immediately, ignoring the Down Server Process policy that can be in effect.
EVENTS <i>option</i>	-	Y	-	<p>The command has three options:</p> <ul style="list-style-type: none"> ♦ EVENTS LIST: Lists all registered events, including third-party events. ♦ EVENTS STATUS: Gives the status of each event. ♦ EVENTS FIRE <i>event_ID</i>: Allows you to manually run an event.
EXIT	-	Y	-	<p>Closes the current command prompt's Java* software. For example, if given at the Subscriber prompt, the Subscriber's Java software is closed.</p>
EXITALL	-	Y	-	<p>Closes the current command prompt's Java and native software.</p>
LISTPLUGINS	-	Y	-	<p>Lists the current ZfS plug-ins.</p>
PACKAGE <i>option</i>	-	Y	-	<p>You can do the following for the software packages installed on the server:</p> <ul style="list-style-type: none"> ♦ PACKAGE LIST: Lists the currently installed software packages. This is useful for knowing which packages can be rolled back and the order that they'll be rolled back, which is the reverse order in which they finished installing, not the order they started installing. ♦ PACKAGE PROCESS <i>full_package_path</i>: Use this to manually install a software package. ♦ PACKAGE ROLLBACK: Automatically rolls back (uninstalls) the most recently installed software package. For example, you installed three software packages on a server (Package1, Package2, and Package3), and Package1 was installed first, Package2 second, and Package3 last. If you want to roll back Package2, you need to first roll back Package3. To do so, type <code>package rollback</code> at the server console once for Package3, then again for Package2. <p>The software package installation order is not guaranteed, because the order is determined by when a package has finished processing. Therefore, the installation order might be Package2, Package1, Package3 when using the Package Rollback command. This order is shown by the Package List command.</p>
POLICY or POLICY LIST	-	Y	-	<p>Lists the effective server policies. Each policy listed has a corresponding policy number for reference when using the POLICY ENFORCE command.</p>

Command	M	Z	T	Description
POLICY ENFORCE <i>policy_number</i>	-	Y	-	Used to manually enforce a specific policy. The <i>policy_number</i> can be found using the POLICY LIST command. This is useful for enforcing a policy ahead of its schedule. However, you will usually use POLICY REFRESH first to ensure you are enforcing the most recent changes.
POLICY ENFORCE ALL	-	Y	-	Used to manually enforce all effective policies, such as after doing a POLICY REFRESH.
POLICY EVENTBASED	-	Y	-	Lists the event-based policies.
POLICY PLUGINS	-	Y	-	Lists the current policy enforcers and the current event handlers.
POLICY REFRESH	-	Y	-	Refreshes only the server's policies and schedules, as required (unlike the REFRESH command, which refreshes policies and undoes any changes made to the prompts). After using this command, you should do a POLICY ENFORCE.
POLICY REFRESHONLY	-	Y	-	Refreshes the server's policies, but does not schedule effective policies.
POLICY RESCHEDULEONLY	-	Y	-	Reschedules all current policies according to their schedules. Does not refresh the effective policies.
POLICY SCHEDULES	-	Y	-	Lists all policy schedules that are in effect.
PROMPT	-	Y	Y	Temporarily resets the current prompt. It will revert back to whatever is specified in the Novell eDirectory™ object for the console prompt when the Java process is exited or restarted, or when the REFRESH command is given.
REFRESH	-	Y	-	Manually forces a refresh of a policy, including pending changes to service locations for the current server and temporary changes to ZfS prompts. IMPORTANT: Do not refresh the Distributor more often than every five minutes. The following can need up to five minutes to complete their processes: Distribution building, eDirectory replication, and tree walking (when no Search policy is defined). Used alone, it refreshes only the ZENworks for Servers policy. Use POLICY REFRESH to refresh all policies. Also restarts the current component's Java process by running the DIST.NCF or SUB.NCF file. You can use this to restart Java as well, because these Java processes will restart Java when they are run. Note that changes to TED object properties are not in effect until the related Distributor re-reads eDirectory.

Command	M	Z	T	Description
SETCONSOLELEVEL <i>number</i>	-	-	-	Sets the console message level: 0: No messages 1: Errors 2: Successes & level 1 messages 3: Warnings & level 2 messages 4: Information & level 3 messages 5: Trace information & level 4 messages 6: Developer trace information & level 5 messages
SETFILELEVEL <i>number</i>	-	Y	Y	Sets the file message level: 0: No messages 1: Errors 2: Successes & level 1 messages 3: Warnings & level 2 messages 4: Information & level 3 messages 5: Trace information & level 4 messages 6: Developer trace information & level 5 messages
SHOWSCHEDULE	-	Y	-	Lists the current schedules.
SHOWVARS	-	Y	-	Lists the predefined variables and their values. These variables can be used in Server Software Packages.
STATUS	-	Y	-	Lists the current status of Policy and Distribution Services, including: Base Path Plug-ins Loaded Events Registered Scheduled Items Console Level
TIME	Y	Y	Y	Returns the current date and time that the server is set to.
VERSION	Y	Y	Y	Returns the ZfS version for the ZfS and TED prompts, and the NetWare version for the console's main prompt.

Java Console Commands

The following table lists some useful Java Virtual Machine (JVM*) commands.

Command	Description
java -show	Lists all loaded Java processes.
java -kill <i>nnn</i>	Kills the specified Java process. (<i>nnn</i> represents the Java process number from the <code>java -show</code> listing.)

Command	Description
java -killall	Stops all loaded Java processes; however, it leaves Java loaded.
java -version	Displays the JVM version.
java -exit or unload java	This attempts to unload all Java process, including the JVM. <code>java -exit</code> is the preferred command. This command is required for unloading any native NLM™ files that are called from Java, such as ZENFILE.NLM.

C

Load/Unload Actions

This section is referenced from other sections.

This information is used in several setup steps for the Server Policies (see [Chapter 17, “Server Policies,”](#) on page 455) or Server Software Packages (see [Chapter 18, “Server Software Packages,”](#) on page 493) components.

- ◆ [“Load NLM/Process”](#) on page 623
- ◆ [“Load Java Class”](#) on page 623
- ◆ [“Unload Process”](#) on page 624
- ◆ [“Start Service”](#) on page 624
- ◆ [“Stop Service”](#) on page 624

Load NLM/Process

For all supported platforms.

If you select an NLM™ to be loaded by the software package, and the NLM is already running on the target server, the package installation will fail and will be rolled back (if rollback is enabled).

You can make sure that an NLM is not already loaded when you are including it in the software package by adding an unload option for that NLM before adding the load option—but only if this NLM does not require user input from the keyboard to unload it.

Filename: This must be the exact name, including the full path to the executable, unless the path to the file is a system path variable. For NLM files, including the .NLM extension.

Parameters: Include any command line parameters for the NLM or process being run.

Wait for this Process to Terminate before Continuing: You can check this option for an NLM or process that will terminate itself. It must terminate within 10 minutes or the whole loading process will fail. By default, this option is dimmed.

Load Java Class

For all supported platforms.

Filename: This must be the exact name. The .CLASS extension is not necessary.

Parameters: Include any command line parameters for the Java application being run.

JVM Parameters: Include any parameters for the Java machine.

Wait for this Process to Terminate before Continuing: You can check this option for a Java application that will terminate itself. There is no time limit. It will wait as long as the application is running. By default, this option is dimmed.

Unload Process

For all supported platforms.

If the NLM requires intervention to unload, you must remember to unload it manually before trying to install the software package.

Filename: This must be the exact name (the path is not required). Because many NLM files require user input to unload, their unloading cannot be automated.

Wait for this Process to Unload before Continuing: You can check this option for a process that will unload itself. By default, this option is dimmed.

Start Service

For Windows* servers only.

Service Name: This must be the exact name.

Wait For This Service to Finish Running Before Continuing: You can check this option for a service that will start itself. By default, this option is dimmed.

Stop Service

For Windows servers only.

Service Name: This must be the exact name.

Wait For This Service to Stop Before Continuing: You can check this option for a service that will stop itself. By default, this option is unchecked.

D

Requirements for Server Software Packages

This section is referenced from other sections.

This information is used in several setup steps for software packages. For more information, see [Chapter 18, “Server Software Packages,” on page 493](#).

IMPORTANT: By selecting a requirement, you are prescribing that it must be met to allow the software package or package component to be installed.

Requirement	Description
Operating System	The operating system (OS) requirements for running the files in the software package, including both the OS the files need for running and whether the target server has that OS.
Memory (RAM)	The minimum RAM required for running the files in the software package. If the target server does not meet that minimum, the software package will not be distributed to it.
Disk Space	The minimum free disk space required for installing the files on the target server. If the target server does not meet that minimum free space, the software package will not be distributed to it.
SET Commands	Which NetWare® SET commands you want specifically configured on the target server for the software package.
Registry	The registry changes that can be required on the target server for the files in the software package. For information on configuring individual registry entries, see Appendix E, “Registry Entries for Server Software Package Components,” on page 629 .
File	Indicates whether a file on the target server should exist or have a certain date.
PRODUCTS.DAT	Changes to PRODUCTS.DAT that the software package requires. Usually, the changes are to update the versions of the software on the server from the contents of the software package. The PRODUCTS.DAT file is used to determine which software and which version exist on the server.

Operating System

You can require the server to have a certain operating system before installing the software package.

To configure the server operating system requirement:

- 1 With the operating system requirement selected, select the server’s platform.

Available platforms are NetWare, Windows, Linux*, and Solaris*.

2 Select the version relationship:

- Any
- Less Than
- Less Than or Equal To
- Equal To
- Greater Than
- Greater Than or Equal To

3 If you select an option other than Any for the Version field, fill in the Major, Minor, and Revision fields according to the information in the following table:

Operating System Version	Major	Minor	Revision
NetWare 5.1 + SP1	5	1	1
NetWare 5.1 + SP2	5	10	2
NetWare 5.1 + SP5	5	10	5
NetWare 5.1 + SP6	5	10	6
NetWare 6 + SP2	6	0	2
NetWare 6 + SP3	6	0	3
Windows NT*	4	0	N/A
Windows 2000	5	0	N/A
Linux (Red Hat* 7.1, 7.2, and 7.3)	2	4	2 or higher (use the <code>uname -a</code> command to determine the exact revision number)
Linux (Red Hat 8)	2	4	18 or higher (use the <code>uname -a</code> command to determine the exact revision number)
Solaris 8	5	8	N/A

The Major and Minor fields are for the upper version limit. The Revision field is for the required service pack revision.

Memory (RAM)

To configure the server memory requirement:

1 With the memory requirement selected, select the condition:

- Less Than
- Less Than or Equal To
- Greater Than
- Greater Than or Equal To

2 Enter the size in megabytes of RAM for the condition selected.

Disk Space

To configure the disk space requirement:

- 1 With the disk space requirement selected, select the root location.

The two options are SYS Volume and Volume. To conserve disk space usage on NetWare servers, do not select the SYS: volume if you have other volumes with available disk space.

Examples of locations you can enter:

NetWare:

SYS:

DATA:

Windows:

C:\

\\MyServer\Data\ (*shared folder*)

Linux or Solaris:

/

/usr

/usr/data

/usr/data

/etc

/mnt/files

For Linux and Solaris servers, it is any path that identifies a disk partition.

- 2 If you selected Volume, enter the volume's name.

- 3 Select the condition:

Less Than

Less Than or Equal To

Greater Than

Greater Than or Equal To

- 4 Enter the free disk space needed in megabytes for the condition selected.

SET Commands

When adding SET commands, the SET Commands Wizard is automatically run.

To configure the SET commands requirement:

- 1 With the SET commands requirement selected, enter the name of the SET command.

- 2 Enter the SET command's value.

Registry

You can require certain entries to exist in the registry before installing the software package.

To configure the registry requirement:

- 1 With the registry requirement selected, select the Entry Type:

Key

Name

Data

- 2 For both entry types Key and Name, select if it Exists or Does Not Exist.

or

For the entry type Data, select if it Equals or Does Not Equal.

- 3 Enter the text for the Key, Name, or Data (depending on which you selected in [Step 1](#)).

Make sure you add the two backslashes to the beginning of the Key. For example, \\HKEY_LOCAL_MACHINE\software\...

IMPORTANT: The % symbol is not valid in NetWare registry names.

File

To configure the file requirement:

- 1 With the file requirement selected, enter the name.

Include the file's full path.

- 2 Select the required file status:

File Exists

File Does Not Exist

Date Is

PRODUCTS.DAT

WARNING: Modifying the PRODUCTS.DAT file could prevent something from running or being installed on the NetWare server. Never modify any entries supplied by Novell®.

To configure the PRODUCTS.DAT requirement:

- 1 With the PRODUCTS.DAT requirement selected, enter the name of item in the .DAT file.

IMPORTANT: Names are case sensitive.

The item is the ID of the product in the .DAT file.

- 2 Enter the version text that corresponds with the item selected in [Step 1](#).
- 3 Select whether the version Contains, Begins With, or Matches the version specified in [Step 2](#).
- 4 Enter the description text that corresponds with the item selected in [Step 1](#).
- 5 Select whether the description Contains, Begins With, or Matches the description entered in [Step 4](#).

E

Registry Entries for Server Software Package Components

This section is referenced from other sections.

The following information is used in several setup steps for software packages. For more information, see [“Registry Settings” on page 518](#).

The NetWare® or Windows registry entries you can change are keys, value names, and value data. You can select keys and value data types for entering changes, and you can enter the corresponding value names when you select one of the types.

In all cases, you must enter the exact key name or value name that is expected in the registry, as well as the correct data values.

The registry settings under HKEY_LOCAL_MACHINE are the only ones you can configure using a software package.

The following registry entries can be changed when you install a software package:

- ◆ [“Key” on page 629](#)
- ◆ [“Binary” on page 630](#)
- ◆ [“Expand String” on page 630](#)
- ◆ [“\(Default\)” on page 630](#)
- ◆ [“DWord” on page 631](#)
- ◆ [“Multi-Value String” on page 631](#)
- ◆ [“String” on page 631](#)

Key

Keys create the paths to the various registry entries. For example, HKEY_LOCAL_MACHINE is a registry key at the root level, and HARDWARE is a key directly under it. The keys are displayed with folder icons in tree fashion. You can click the plus or minus signs to expand or compress the tree structure.

In the box where the HKEY_LOCAL_MACHINE key is displayed, you need to use the Key registry entry to create the path to where the registry changes will be placed.

To configure a Key entry:

- 1** In the box displaying your key tree, click the location where you want the key entered.
- 2** Click Key from the drop-down box > click Add.

New Key #1 is displayed.

- 3** Change the default key name to the key name that you need.
IMPORTANT: After typing the new key name, you must press Enter to save the change.
- 4** Select a condition for making the registry change:
 - Create
 - Delete
- 5** To apply the setting to all subordinate keys, click Apply To All.

Binary

A value data type that is a list of hexadecimal numbers, such as:

```
d0 04 72 6e
```

You must first use the Key registry setting option to create the path to the key that will hold the Binary information.

To configure a Binary entry:

- 1** In the box displaying your key tree, click the location where you want the binary data entered.
- 2** Click Binary from the drop-down box > click Add.
New Value #1 is displayed.
- 3** Change the default Binary name to the name that you need.
- 4** Select a condition for making the registry change:
 - Create
 - Delete
- 5** Enter the binary data.

The Data box is a hexadecimal editor. There are three unlabeled columns:

First: Binary counter of the number of hexadecimal characters, beginning with 0000.

Second: Hexadecimal data, eight entries per row.

Third: Plain text ASCII characters corresponding to the hexadecimal data.

You can enter data in either the second or third column. As you enter data in one the second (hexadecimal) column, the corresponding characters are displayed in the third (text) column, and vice versa.

Expand String

NetWare only. Currently not supported.

(Default)

This is usually the first data entry for a key.

You must first use the Key registry setting option to create the path to the key that will hold the (Default) entry.

To configure a (Default) entry:

- 1** In the box displaying your key tree, click the location where you want the (Default) entry made.
- 2** Click (Default) from the drop-down box > click Add.
(Default) is displayed.
- 3** With the (Default) entry selected, select a condition for making the registry change:
 - Create
 - Delete
- 4** Enter a string in Data.

DWord

DWords are based on hexadecimal code that is represented in Double WORD format. For example:

0x00100022

You must first use the Key registry setting option to create the path to the key that will hold the DWord information.

To configure a DWord entry:

- 1** In the box displaying your key tree, click the location where you want the DWord entry made.
- 2** Click DWord from the drop-down box > click Add.
New Value #1 is displayed.
- 3** Change the default DWord name to the name that you need.
- 4** Select a condition for making the registry change:
 - Create
 - Delete
- 5** Enter the DWord string in Data.

Multi-Value String

NetWare only. Currently not supported.

String

String values are easy-to-read sequences of words or numbers within quote marks.

You must first use the Key registry setting option to create the path to the key that will hold the String information.

To configure a String entry:

- 1** In the box displaying your key tree, click the location where you want the String data entered.
- 2** Click String from the drop-down box > click Add.
New Value #1 is displayed.
- 3** Change the default String name to the name that you need.

4 Select a condition for making the registry change:

Create

Delete

5 Enter the string in Data.

F

Using Server Software Packages to Delete Directories on Servers

If you want to delete certain directories from a number of different network servers (NetWare[®], Windows, Linux, and Solaris), you normally do not have an automated method for performing this task. However, if you are using Novell[®] ZENworks[®] for Servers (ZfS) 3.0.2 Policy and Distribution Services, the Server Software Packages feature of ZfS provides the capability for you to delete specified directories from any Subscriber server's file system.

To automate the deletion of specified directories on multiple servers, you will first set up path variables (if necessary), create a Server Software Package in its namespace in ConsoleOne[®], compile the software package, then distribute the package using Tiered Electronic Distribution (TED). No further user intervention will be required.

Do the following in order to create a software package that will delete specified directories on a server:

1. [“Setting Up Variables for Use With the Server Software Package” on page 633](#)
2. [“Creating the Server Software Package” on page 634](#)
3. [“Creating and Configuring the Server Software Package Component” on page 634](#)
4. [“Compiling the Server Software Package” on page 636](#)
5. [“Manually Testing that the Directories Have Been Deleted” on page 636](#)
6. [“In Summary” on page 636](#)

Setting Up Variables for Use With the Server Software Package

Before you create the software package, you must set up the variables in your Subscriber objects' properties if you will be using variables in paths (for instance, if your target servers have different operating systems, like NetWare and Windows).

- 1** Identify the directories to be deleted:
 - 1a** Identify the root of the path, such as its volume name (NetWare), drive letter (Windows), or /usr (for Linux and Solaris). For example, DATA.
 - 1b** Identify the rest of the path, including the parent directory to the directories to be deleted, such as ZENWORKS\PDS\TED\DIST where DIST is the parent directory.
 - 1c** Identify the directories to be deleted, such as OldDist.TED.ZfS3.Novell.

The resulting full path and directory to be deleted would be:

```
DATA:\ZENWORKS\PDS\TED\DIST\OldDist.TED.ZfS3.Novell
```

You might have varying path elements from server to server. You should use variables as necessary to allow for those differences (see [Step 2](#) and [Step 3](#)).

- 2 In ConsoleOne, create a variable to represent DATA:, D:, or /usr for each Subscriber where the directories to be deleted reside, such as DELETEDDIRROOT.

If you name a directory to be deleted that does not exist on a target server, nothing will be done for that directory on that server.

You can also define variables globally using the Tiered Electronic Distribution policy, where you would define the default value for a variable and allow the exceptions to be defined in the applicable Subscriber objects' properties.

- 3 In ConsoleOne, create a Subscriber variable to represent where any path elements are different.

If you have an extra directory between the root of the drive on your Windows servers (such as ZFS3), you will need to create a variable on all of your target Subscriber servers for that part of the path. For example, if your Windows servers have ZFS3\ZENWORKS at the root of the D: drive, and your NetWare servers have only ZENWORKS at the root of the DATA: volume, create a variable (such as %TARGET%) to represent ZFS3\ZENWORKS on your Windows Subscribers and ZENWORKS on your NetWare Subscribers.

- 4 Repeat [Step 2](#) and [Step 3](#) as necessary.

Creating the Server Software Package

- 1 In the left pane in ConsoleOne where the Zfs 3.0.2 snap-ins have been installed, right-click the Server Software Packages namespace.
- 2 Click File > New > Software Package. to start the Create New Server Software Package Wizard.
- 3 Click Next.
- 4 Enter a name for the software package, such as Delete Old Directories.
- 5 Specify a path and filename for the software package template file (.SPK), such as C:\TEMP\DELETEDIRS.SPK.

IMPORTANT: If you save your .SPK file to a network server, use a UNC path so that you will still have access to that software package file if your drive letters change.

You can also save your .SPK files to a workstation or server, because the .SPK file sizes do not become large. For this particular type of software package (where you are only giving instructions for deleting directories and are not compiling data files), the .CPK (compiled software package) version will be similar in size. Therefore, for management purposes, you may want to save these .SPK files and their corresponding .CPK files in the same location, which can be on a workstation or server.

- 6 Click Finish.
- 7 If necessary, click the plus sign to expand the Server Software Package namespace to view the new package.

Unless otherwise instructed, steps in the subsequent sections should be performed from the same instance of ConsoleOne you used in the above steps, because your .SPK files will be accessible from there.

Creating and Configuring the Server Software Package Component

- 1 Right-click the software package object that you just created and select New Component.
- 2 Enter a name for the component, such as Delete Directories.

- 3** If necessary, click the plus sign to expand the Server Software Package object.
- 4** Right-click the component and select Properties.
- 5** Click the Copy File tab.
- 6** Click the drop-down list button next to the Add button and select Add File Group.
- 7** Click Add.
- 8** Enter a name for the file group, such as Delete Working Directories.
- 9** In the Group Target Path field, enter the name of the variable that you created containing the location of the directories to be deleted, and add any path information that is not contained in the variable; however, do not enter the name of the directory to be deleted as part of that path.

For example, if the location for the directories to be deleted is the same for all target servers, enter the actual volume (NetWare) or drive (Windows) with the path information (which can also contain variables).

However, if you need to use variables because the server operating systems are different, then enter the variable name (within the % symbols) plus the full path (which can also contain variables) to the directory just above the directories to be deleted. For example, %DELETEDDIRROOT% (variable name) and %TARGET%\PDS\TED\DIST (full path to the parent directory of the directories you want to delete).

IMPORTANT: When using variables, the path you enter must be the directory containing the directory to be deleted. In Step 11 you will add the actual directory names to be deleted.

- 10** Click OK to exit the dialog box.
- 11** Click the drop-down list button again and select Add Directory.
Make sure you first select the tree item under which you want to add this directory.
- 12** Click Add.
- 13** To change the name (“Directory”) that defaults in the tree structure to the actual directory name that you want deleted (such as OldDist.TED.Zfs3.Novell), edit the directory name and press the Enter key to save the change.
If you do not press the Enter key, “Directory” will be displayed again. The Rename button allows you to edit the directory name.
- 14** Click the drop-down list button next to the Copy Mode combo box and select Delete.
- 15** Click Apply.
- 16** Repeat **Step 10** through **Step 15** for each directory you want this software package to delete using this component’s file group.

You can start at **Step 6** to add other file groups, or from **Step 1** to add a new component. You might want to repeat from these steps if you cannot add all of your directories to be deleted under the file group that you created in **Step 6**.

- 17** When finished configuring the software package component, click OK or Close.

Using the examples from the above steps, you would have entered:

`%DELETEDDIRROOT%`

and

`%TARGET%\PDS\TED\DIST`

and

`OldDist.Zfs3.TED.Novell`

in order to delete the directories having these paths:

```
DATA:\ZENWORKS\PDS\TED\DIST\OldDist.TED.Zfs3.Novell  
D:\ZFS3\ZENWORKS\PDS\TED\DIST\OldDist.TED.Zfs3.Novell
```

Compiling the Server Software Package

You now have a .spk file that serves as the template for what you want to delete. You need to compile this .SPK file into a .CPK file.

- 1 Right-click the software package, such as Delete Old Directories.
- 2 Select Compile to start the Compile Software Package Wizard.
- 3 Click Next on the first page of the wizard.
- 4 Enter the full path and filename for the .CPK file that you will be generating.

IMPORTANT: Do not use the .SPK extension for this filename, or your template file could be overwritten by its compiled version if they are stored in the same location. This would prevent you from making further edits to the software package. You can use the same filename, such as DELETEDIRS, but you should use only the .CPK filename extension.

- 5 Click Next, then click Finish.

Manually Testing that the Directories Have Been Deleted

The software package is now ready for sending as a Software Package Distribution. However, for testing, you can manually process the software package on one of the target servers to determine that the directories were deleted as intended.

- 1 On a server where you want to delete a directory, create a directory that is contained in your software package (such as OldDist.TED.Zfs3.Novell) under ZENWORKS\PDS\TED\DIST.
- 2 Copy the .CPK file (for example, DELETEDIRS.CPK) to the TEMP directory on that server.
- 3 At the server's Zfs console prompt, enter the PACKAGE PROCESS command to process the software package.

For example, if it was a NetWare server, at the Zfs prompt you would enter:

```
package process data:\temp\deletedirs.cpk
```

Zfs will process the package and report that it has finished processing. Check the server's file system to see that the OldDist.TED.Zfs3.Novell directory, or the directories you specified, were deleted.

In Summary

After you are satisfied with the result of your test, you can distribute the DELETEDIRS.CPK file using TED to all your target Subscriber servers with your new Software Package Distribution in order to delete directories on your Subscriber servers' file systems.

G

Documentation Updates

This section contains information on documentation content changes that have been made in the *Administration* guide for Policy and Distribution Services since the initial release of Novell® ZENworks® for Servers (ZfS) 3. This information will help you to keep current on updates to the documentation.

If you have purchased ZfS 3.0.2 and have not used or installed ZfS 3 or ZfS 3 SP1, you do not need to review this section.

All changes that are noted in this section were also made in the documentation. The documentation is provided on the Web in two formats: HTML and PDF. The HTML and PDF documentation are both kept up-to-date with the documentation changes listed in this section.

The documentation update information is grouped according to the date the documentation updates were published. Within a dated section, the changes are alphabetically listed by the names of the main table of contents sections for Policy and Distribution Services.

If you need to know whether a copy of the PDF documentation you are using is the most recent, the PDF document contains the date it was published on the front title page or in the Legal Notices section immediately following the title page.

The documentation was updated on the following dates:

- ◆ “November 8, 2004” on page 637
- ◆ “July 29, 2004” on page 638
- ◆ “March 10, 2004” on page 639
- ◆ “June 27, 2003” on page 639
- ◆ “April 15, 2003” on page 641
- ◆ “December 19, 2002” on page 643
- ◆ “September 27, 2002” on page 646
- ◆ “June 6, 2002” on page 647
- ◆ “May 17, 2002” on page 648

November 8, 2004

Throughout the guide, graphics have been updated with font changes and newer icons. No notable content changes were made to any graphics.

Updates were made to the following sections. The changes are explained below.

- ◆ **Desktop Application Distribution**

Desktop Application Distribution

The following changes were made in this section:

Location	Change
Chapter 19, “Desktop Application Distribution,” on page 525	Added new sections, especially for concepts, and updated the existing sections.

July 29, 2004

Updates were made to the following sections. The changes are explained below.

- ◆ [Server Console Commands](#)

Tiered Electronic Distribution

The following changes were made in various sections in the guide:

Location	Change
“Manually Importing and Exporting Distributions” on page 412	Updated this section, including adding a new step that initiates the imported Distribution so that it is extracted and its status can be viewed in iManager.

Server Policies

The following changes were made in various sections in the guide:

Location	Change
“Tiered Electronic Distribution” on page 471	In Step 4, rewrote the text for the Filename field, clarifying which agents log Tiered Electronic Distribution information.
“ZENworks for Servers” on page 485	In Step 4, rewrote the text for the Filename field, clarifying which agent logs policy and software package information.
“Creating Custom Log Files Using Policies” on page 486	Added this new section explaining how to create custom logging for reporting information.

Server Console Commands

The following changes were made in various sections in the guide:

Location	Change
“ZfS Console Commands” on page 617	Reworded the Description to read: Lists the predefined variables and their values. These variables can be used in Server Software Packages.

March 10, 2004

Updates were made to the following sections. The changes are explained below.

- ◆ [Tiered Electronic Distribution](#)

Tiered Electronic Distribution

The following changes were made in various sections in the guide:

Location	Change
“Configuring Subscribers” on page 420	<p>Added the following paragraph:</p> <p>If a Subscriber object is inadvertently deleted, you can re-create it in ConsoleOne. However, the revision number of the new Subscriber object will be less than its revision number in the ted.cfg file. Therefore, the Subscriber cannot accept any updates to its configuration, because the lower revision number causes it to assume that the configuration data is older than what it has. To resolve this problem, delete the ted.cfg file on the Subscriber server, and the next time a Distribution is sent to the Subscriber, a new configuration is accepted, and a new ted.cfg file created.</p>

June 27, 2003

Updates were made to the following sections. The changes are explained below.

- ◆ [Policy and Distribution Services](#)
- ◆ [Tiered Electronic Distribution](#)
- ◆ [Server Policies](#)
- ◆ [Security in Policy and Distribution Services](#)

Policy and Distribution Services

The following changes were made in various sections in the guide:

Location	Change
A general change throughout the guide.	<p>The following note was changed in all locations where it appeared in the guide. The previous recommendation was to not refresh the Distributor more often than every three minutes. This has been corrected to every five minutes, and the processes involved were added to the notes to explain why five minutes is needed.</p> <p>IMPORTANT: We recommend the Distributor's Refresh schedule be daily, unless changes to Distributions warrant a more frequent refresh. However, do not refresh the Distributor more often than every five minutes. The following can need up to five minutes to complete their processes: Distribution building, eDirectory replication, and tree walking (when no Search policy is defined).</p>

Tiered Electronic Distribution

The following changes were made in this section:

Location	Change
Chapter 16, “Tiered Electronic Distribution,” on page 371	This whole section has been reorganized, causing many sections to be moved, some renamed, some incorporated within others, and some new sections added. This new organization should improve your ability to find information on TED.
“Setting Subscribers’ Extract Schedules” on page 348	Rewrote this section. It had previously indicated that you could modify the schedules of multiple Subscriber objects using the Properties of Multiple Objects menu option, which cannot be done. The Schedule tab does not display when selecting multiple Subscriber objects.
“Updating Subscriber Configurations” on page 423	Added this new section that explains how a Subscriber server receives updates to its TED configuration file.
“Understanding External Subscribers” on page 426	<p>Altered the External Subscriber graphic by removing the Parent Subscriber section, which was misleading. External Subscribers can receive Distributions either directly from a Distributor or via a parent Subscriber in the Distributor’s distribution hierarchy.</p> <p>Also added information to the end of the following paragraph for clarification:</p> <p>The External Subscriber object’s properties lists the Channels it can receive Distributions from. An External Subscriber cannot be a parent Subscriber itself, though if it has a parent Subscriber, both the External Subscriber’s and parent Subscriber’s objects must reside in the same tree. An External Subscriber can receive Distributions directly from a Distributor, without using a parent Subscriber, or it can receive Distributions via a parent Subscriber in the Distributor’s distribution hierarchy.</p>
“Subscriber Software Configuration and Trusted Trees” on page 428	This new section documents the issues related to trusted trees, which come into play when using External Subscriber objects.
“External Subscriber, Multiple Distributors, and Multiple Trees” on page 431	<p>Added the parenthetical defining a trusted tree in the following paragraph:</p> <p>In this example, each tree has a Distributor. Server_4 receives its configuration information from Distributor_B (Server_3) in its trusted tree (the tree where the Subscriber’s object resides, not the tree where its associated External Subscriber object resides; in this case, the trusted tree is Tree_B). Therefore, a TEDNODE.PROPERTIES file is not needed for Server_4.</p>
“Editing the TEDNODE.PROPERTIES File” on page 451	<p>The following paragraph was removed. This file is no longer contained on the CD, because with Zfs 3 it is installed on every server where the Subscriber software is installed.</p> <p>A sample of the TEDNODE.PROPERTIES file is located at the root of the <i>ZENworks for Servers Companion</i> CD or under the ZENWORKSFORSERVERS directory of the <i>ZENworks 6 Companion 1</i> CD.</p>

Location	Change
“Editing the TEDNODE.PROPERTIES File” on page 451	The first paragraph was reworded to correct the fact that the TEDNODE.PROPERTIES file is now installed on servers and no longer available on the CD.

Server Policies

The following changes were made in this section:

Location	Change
Chapter 21, “Scheduling,” on page 565	Rearranged the contents in this section, and added steps in the Scheduling the TED Objects section for how to schedule each TED object (Distributor, Distribution, Channel, and Subscriber).
“Understanding Scheduling in Policy and Distribution Services” on page 565	Moved this section from the Tiered Electronic Distribution section.

Desktop Application Distribution

The following changes were made in this section:

Location	Change
“Requirements” on page 538	Added the following bullet: <ul style="list-style-type: none"> For a Desktop Application Distribution that contains a large amount of registry setting information, you can receive a Java out of memory error when the Distribution is being extracted. To prevent this, edit the TED.NCF file on the Subscriber server and change the memory variable on the last line from 128 to 256. Then the Distribution should extract.

Security in Policy and Distribution Services

The following changes were made in this section:

Location	Change
“Important Points about Certificates” on page 551	Added the following bullet to this section: <ul style="list-style-type: none"> ConsoleOne copies the certificate files to Subscriber servers. Therefore, the client software on the workstation running ConsoleOne must have access to the Subscriber servers' file systems. For Windows Subscriber servers, the Domain and Workgroup rights on the workstation must be set up to facilitate automatic certificate copying. Otherwise, a 1204a error will be given.

April 15, 2003

Updates in this section correspond to the release of the ZfS 3.0.2 CDs and ZfS 3 SP2.

Updates were made to the following sections. The changes are explained below.

- ◆ [Configuring Policy and Distribution Services](#)
- ◆ [Tiered Electronic Distribution](#)
- ◆ [Server Software Packages](#)
- ◆ [Security in Policy and Distribution Services](#)
- ◆ [Appendixes: Server Console Commands](#)
- ◆ [Appendixes: Requirements for Server Software Packages](#)

Configuring Policy and Distribution Services

The following changes were made in this section:

Location	Change
“Installing NICI 2.4” on page 341	Revised this section.

Tiered Electronic Distribution

The following changes were made in this section:

Location	Change
“TED Issues” on page 442	Moved this section to a higher section level to give it more visibility.
“Changing DNS Names or IP Addresses for TED Servers” on page 446	Added this new section concerning what to do to maintain the distribution processes after changing a Distributor or Subscriber server’s DNS name or IP address.

Server Software Packages

The following changes were made in this section:

Location	Change
“Determining the Installation Order of Software Packages” on page 495	Rewrote this section to add more information concerning the installation order of software packages that are contained in the same Distribution, especially how it affects rollback.
“Distributing Software Packages to a Cluster” on page 498	Added this new section concerning issues with distributing Server Software Packages to clustered servers.
“Registry Settings” on page 518	Added the following paragraph under Step 1: HKEY_LOCAL_MACHINE is a Windows registry key. For NetWare, HKEY_LOCAL_MACHINE is also recognized by ZFS as the equivalent to My Server. Therefore, you can use this key for editing both NetWare and Windows registries.

Security in Policy and Distribution Services

The following changes were made in this section:

Location	Change
“ConsoleOne User Rights and Certificate Copying” on page 552	Added this new section concerning certificate copying and ConsoleOne user rights, which are automatic for NetWare Subscribers, but must be set for Windows Subscribers.

Reporting

The following changes were made in this section:

Location	Change
“Creating Customized Reports” on page 605	Moved this section from Chapter 23, “ZENworks Database,” on page 587 to Chapter 24, “Reporting,” on page 599 and renamed it (previously titled “Database Contents”).
“Default Sybase Database User ID and Password” on page 605	Added this new section to provide the default user ID and password for accessing the Sybase database file (ZFSLOG.DB) that ships with ZfS.

Appendixes: Server Console Commands

The following changes were made in this section:

Location	Change
Appendix B, “Server Console Commands,” on page 617	Added information in the explanation of the Package command concerning how rollback is affected by the fact that a specified software package processing order is not guaranteed, because the order packages are listed depends on when they finished processing, not when they started processing.

Appendixes: Requirements for Server Software Packages

The following changes were made in this section:

Location	Change
“Operating System” on page 625	Because ZfS 3.0.2 minimum requirements have changed, updated the table under Step 3 for additional operating systems concerning major, minor, and revision numbers.

December 19, 2002

Updates were made to the following sections. The changes are explained below.

- ◆ [Configuring Policy and Distribution Services](#)
- ◆ [Tiered Electronic Distribution](#)

- ◆ [Server Software Packages](#)
- ◆ [Desktop Application Distribution](#)
- ◆ [Variables](#)
- ◆ [ZENworks Database](#)
- ◆ [Appendixes: Using Server Software Packages to Delete Directories on Servers](#)

Configuring Policy and Distribution Services

The following changes were made in this section:

Location	Change
"Desktop Application" on page 322	<p>Added a new second paragraph concerning inter-tree distributions of Application objects:</p> <p>You can distribute Desktop Application Distributions to a Subscriber server on a tree different from the Distributor server. However, this recipient server's Subscriber object and NCP object must reside on the same tree. The Desktop Application Distribution can be sent to such a server on another tree using an External Subscriber object on the Distributor's tree.</p>
"Desktop Application" on page 322	<p>Added the following new fourth paragraph:</p> <p>The rebuild of a Desktop Application Distribution can also be triggered by any change to the Application object that changes its Revision value. In this case, the Desktop Application Distribution is built as a delta that contains only the files that have changed.</p>
"Desktop Application" on page 322	<p>Added the following item in the worksheet entry table.</p> <p>Under item 3 and item 20, indicate that you will have Desktop Application Distributions, and therefore each server that will be receiving Desktop Application Distributions must have its Subscriber object and NCP Server object on the same tree.</p>
"Software Package" on page 322	<p>Added a new note concerning the order of .CPK file installations when you have multiple software packages in one Distribution:</p> <p>IMPORTANT: The order that the .CPK files are applied on a server is not guaranteed, and .CPK files contained in one Distribution that may start in a certain order might not all finish in that same order. Therefore, place each .CPK file in its own Distribution if you want them to be installed in a particular order and use Distribution scheduling to determine the order. For more information, see "Forcing the Software Package Distribution Order" on page 495.</p>

Tiered Electronic Distribution

The following changes were made in this section:

Location	Change
"Maximum Revisions" on page 396	With reference to the Maximum Revisions field, the following paragraph was added: If you enter 1, the Delete Previous Revision Before Receiving Next field becomes accessible. This allows you to control disk space by only maintaining one copy of a Distribution on the server's file system.
"Deleting a Distributor Object and How Its Distributions Are Affected" on page 392	Added this new section to clarify what happens to Distributions when the related Distributor object is deleted from eDirectory.

Server Software Packages

The following changes were made in this section:

Location	Change
"Determining the Installation Order of Software Packages" on page 495	Added this new section concerning the installation order of software packages that are contained in the same Distribution, which order is not guaranteed.
"What Are My Software Package Management Options?" on page 500	Added this new section to explain management issues related to software package files.
"Setting Up Multiple-Workstation Management for Server Software Packages" on page 506	Added this new section for how to set up software package management from multiple workstations.

Desktop Application Distribution

The following changes were made in this section:

Location	Change
"Rebuilding Desktop Application Distributions" on page 545	Added this new section with information about how a rebuild of a Desktop Application Distribution is triggered, and what is contained in the rebuilt Distribution.

Variables

The following changes were made in this section:

Location	Change
"Resolution of Variable Names" on page 581	Updated this section to clarify how variable names are resolved. Included clarification that for Server Software Packages, its variable settings override the Subscriber's variable settings.

ZENworks Database

The following changes were made in this section:

Location	Change
“Creating Customized Reports” on page 605	Added the following paragraph concerning reporting on TED objects: However, for TED objects such as a Subscriber or the External Subscriber, you should use ZENworks reporting options (see Chapter 24, “Reporting,” on page 599) or iManager (Chapter 15, “Novell iManager,” on page 359) for determining the status of Distributions or policies.

Appendixes: Using Server Software Packages to Delete Directories on Servers

The following changes were made in this section:

Location	Change
Appendix F, “Using Server Software Packages to Delete Directories on Servers,” on page 633	Added this new section that explains how to use software packages to delete directories on network servers.

September 27, 2002

Updates were made to the following sections. The changes are explained below.

- ◆ [Policy and Distribution Services](#)
- ◆ [Configuring Policy and Distribution Services](#)
- ◆ [Novell iManager](#)
- ◆ [Security in Policy and Distribution Services](#)

Policy and Distribution Services

The following changes were made in this section:

Location	Change
“Policy and Distribution Services” on page 315	Renamed all of the major sections, moved the Novell iManager section up from sixth to second position, and moved the Variables section near the end of the Policy and Distribution Services section.

Configuring Policy and Distribution Services

The following changes were made in this section:

Location	Change
Chapter 14, “Configuring Policy and Distribution Services,” on page 317	Reorganized and rewrote this entire section to correspond to the changes in “Installing Policy and Distribution Services on NetWare and Windows Servers” in the <i>Installation</i> guide.

Novell iManager

The following changes were made in this section:

Location	Change
“Managing the Policy/Package Agent from the Remote Web Console” on page 366	You can immediately enforce or remove a specific policy on a Subscriber server.
“Comparing the ZfS Management Role in iManager with ConsoleOne Capabilities” on page 368	You can now create the Desktop Application Distribution in Novell iManager.

Security in Policy and Distribution Services

The following changes were made in this section:

Location	Change
“Handling Invalid Certificates” on page 553	Rewrote this entire section concerning certificates becoming invalid due to changing a DNS name or IP address. This section previously contained two paragraphs. It now contains two subsections with steps.

June 6, 2002

Updates were made to the following sections. The changes are explained below.

- ◆ [Planning a Policy and Distribution Services Configuration](#)
- ◆ [Security in Policy and Distribution Services](#)

Planning a Policy and Distribution Services Configuration

The following changes were made in this section:

Location	Change
“Configuring Distributors in a Mixed eDirectory Environment” on page 328	Added this new section to document configuring Distributors to work in a mixed Novell eDirectory™ environment.

Understanding Security in ZENworks for Servers

The following changes were made in this section:

Location	Change
“Handling Invalid Certificates” on page 553	Replaced the two paragraphs in this section with new information, including two subsections with instructions on what to do when an IP address or DNS name is changed.

May 17, 2002

Updates were made to the following sections. The changes are explained below.

- ◆ [Planning the Configuration](#)
- ◆ [Configuring Policy and Distribution Services](#)
- ◆ [Managing Your Distribution System](#)
- ◆ [Understanding Security in ZENworks for Servers](#)

Planning the Configuration

The following changes were made in this section:

Location	Change
“Determining Whether You Need Encryption Security for Windows Servers” on page 332	NICI must be installed on Windows servers where encrypted Distributions are either built or extracted. A new planning section was added.
“Configuration Planning Worksheet” on page 349	Inserted two new worksheet items at the beginning for installing NICI, causing all other worksheet items to be renumbered.
“The TEDNODE.PROPERTIES File Requirement” on page 430	In the last bullet, corrected the location for where to find the TEDNODE.PROPERTIES file: A sample of the TEDNODE.PROPERTIES file is located on the <i>ZENworks for Servers Companion</i> CD under ZFS\TEDPOL\FILES\TED or the <i>ZENworks 6 Companion 1</i> CD under ZENWORKSFORSERVERS\ZFS\TEDPOL\FILES\TED.
“Multiple Databases” on page 590	Added information in this section concerning multiple databases and Distributor objects using the same context. Only one Service Location Package can be associated with a given context. This section has been rearranged into three subsections.

Configuring Policy and Distribution Services

The following changes were made in this section:

Location	Change
“Installing Additional Distributors, Databases, and Subscribers” on page 337	Clarified database installation: only one can be installed per run of the installation program. Edited several steps and added a final step for repeating the process to install additional databases.

Managing Your Distribution System

The following changes were made in this section:

Location	Change
“Refreshing the Distributor” on page 392	Added this new section to document why and how to refresh the Distributor.
“Tiered Electronic Distribution” on page 471	Added the following paragraph in Step 5 under Value: To ensure that extraction will take place, provide an absolute path to the Subscriber. For example, if the path is only the DATA volume, make sure the colon (:) is included, because it is a necessary part of the full path. Also added this paragraph in: Step 7 under Value in “Configuring Subscribers” on page 420 . Step 5 under Value in “Tiered Electronic Distribution” on page 471 . Step 4 in “Creating Default Variables for All Subscribers” on page 582 . Step 4 in “Creating Variables for a Specific Subscriber” on page 583 . Step 4 in “Using Variables to Control File Extraction” on page 584 .

Automating Server Software Installations and Updates

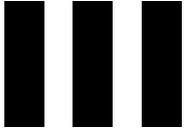
The following changes were made in this section:

Location	Change
“Converting Older Server Software Packages to ZFS 3.0.2” on page 521	Corrected the TID number to 2962260 (was incorrectly listed as 10069358).

Understanding Security in ZENworks for Servers

The following changes were made in this section:

Location	Change
“Security for Inter-Server Communication Across Non-Secured Connections” on page 562	Moved all of the installation information inter-server communications security into the <i>Installation</i> guide. For the new location of the moved information, see “Installing Additional Security for Non-Secured Connections” under Installing ZENworks for Servers in the <i>Installation</i> guide.



Server Inventory

Novell® ZENworks® for Servers (ZfS) Server Inventory enables you to collect hardware and software inventory information from the local and the remote servers of your enterprise. This inventory information is scanned and stored in a database that can be accessed by the network administrator.

From ConsoleOne®, you can view the complete hardware and software inventory of the servers. You can also query the centralized database of the servers.

The Server Inventory documentation contains the following sections:

- ♦ [Chapter 25, “Understanding Server Inventory,” on page 653](#)
- ♦ [Chapter 26, “Setting Up Server Inventory,” on page 661](#)
- ♦ [Chapter 27, “Understanding the Server Inventory Components,” on page 725](#)
- ♦ [Chapter 28, “Understanding the ZENworks for Servers Inventory Database Schema,” on page 787](#)
- ♦ [Chapter 29, “Managing Inventory Information,” on page 811](#)
- ♦ [Chapter 30, “Monitoring Server Inventory Using Status Logs,” on page 843](#)
- ♦ [Chapter H, “Documentation Updates,” on page 849](#)

25 Understanding Server Inventory

Novell® ZENworks® for Servers (ZfS) Server Inventory gathers hardware and software inventory information from the NetWare® 5.1/6 and Windows* NT* 4.0/2000 servers in your enterprise and stores into a centralized database. Using this database, the network administrator can view and query for complete inventory information for the enterprise.

This chapter provides a basic overview of the ZfS Server Inventory service. It contains the following information:

- ♦ “Server Inventory Terminology” on page 653
- ♦ “Overview of Server Inventory Components” on page 654
- ♦ “Understanding the Inventory Scanning Cycle in the Standalone Scenario” on page 656
- ♦ “Understanding Rolling Up Scan Data Across Servers” on page 657

Server Inventory Terminology

The following brief glossary provides basic definitions of Server Inventory terms:

Inventoried server: A server whose hardware and software data you want to scan and maintain in a central repository. To gather complete hardware and software inventory for a server, you must install the Inventory Agent on that server.

Inventory database: A repository of inventory information of all the inventoried servers.

Inventory server: A server where you run the Inventory service. This server can run any other ZfS 3 services also. The Inventory server collects the inventory data from a group of associated inventoried servers and stores it into the Inventory database. If you want to collect the inventory for the Inventory server, you must install the Inventory Agent on that Inventory server.

Database server: A server running Sybase* or Oracle* where your Inventory database is mounted. The database can run on an Inventory server or on a different server.

Management console: A Windows workstation or server running Novell ConsoleOne® with ZfS 3 Server Inventory ConsoleOne snap-ins installed. The management console provides the interface to administer the inventory system.

eDirectory Tree: The Novell eDirectory™ tree consists of eDirectory objects such as multiple levels of organizational units, users, groups, and other network resources. This hierarchical structure is referred to as the eDirectory tree in this document. For more information, see the [Novell eDirectory documentation Web site \(http://www.novell.com/documentation\)](http://www.novell.com/documentation).

Inventory tree: A logical tree depicting the transmission of the inventory information from the inventoried servers and the Inventory servers to the centralized enterprise Inventory database.

Standalone Server: An Inventory server that has an Inventory database and inventoried servers attached to it. There is no roll-up of the inventory information.

Leaf Server: The lowest-level Inventory server in the inventory tree hierarchy. This server has one or more inventoried servers attached to it and can have the Inventory database attached to. This Inventory server collects the inventory information from the inventoried servers attached to it and moves the information to the next-level Inventory server.

Intermediate Server: The staging Inventory server for moving the data from the lower-level Inventory servers up the Inventory server hierarchy. This server can have inventoried servers or the Inventory database attached to it.

Root Server: The highest-level Inventory server in the inventory tree hierarchy. This server has a centralized Inventory database that contains the inventory information of all the lower-level Inventory servers. At the Root Server level, you can view complete inventory information for the entire enterprise. This server can have inventoried servers attached to it.

Inventory site: A single site with a simple network environment of inventoried servers and at least one Inventory server. A site is typically a geographical location. There can be multiple sites your enterprise.

Overview of Server Inventory Components

Before setting up the ZfS inventory deployment, you should understand the inventory components that interact together to perform inventory functions.

ZfS Server Inventory uses the following components:

- ◆ “Inventory Scanners” on page 654
- ◆ “Inventory Components on Inventory Servers” on page 654
- ◆ “Inventory Database” on page 655
- ◆ “Management Console” on page 655

Inventory Scanners

Platform-dependent scanners determine the hardware and software configurations of the inventoried servers. These scanners are located at the inventoried servers. When executed on the inventoried servers, the scanners collect the inventory information and store the scan data as .str files. The .str files are subsequently transferred to the Inventory server and processed.

Using the Server Inventory policy, you can configure the scan settings so you can schedule the scanning on the inventoried servers, enable a software scan, and customize software scanning. From the Inventory Service object, you can specify the location of the scan data files.

Inventory Components on Inventory Servers

The inventory components process the scan data. The following components are Java* programs that work identically on NetWare and Windows NT/2000 Inventory servers:

- ◆ Scan Collector

The Scan Collector collects the .str files and stores them in the scan directory (scandir) at the Inventory server. The .str files are transferred using the XML-RPC protocol.

- ◆ Selector

The Selector processes the .str files and places the files in the dbdir and entmergedir directories.

- ◆ Sender and Receiver

The Sender and the Receiver on the Inventory servers compress the .str files and then transfer the files from the lower-level Inventory servers to the higher-level Inventory servers for roll-up of inventory information. By using the Roll-Up policy, you can configure the next level destination Inventory server for roll-up, and also schedule the roll-up time.

- ◆ Storer

The Storer stores the collected inventory information (.str files) in the Inventory database. By using the Database Location policy, you can configure the properties of the Inventory database object in ZfS and associate the database object to an Inventory server.

Inventory Database

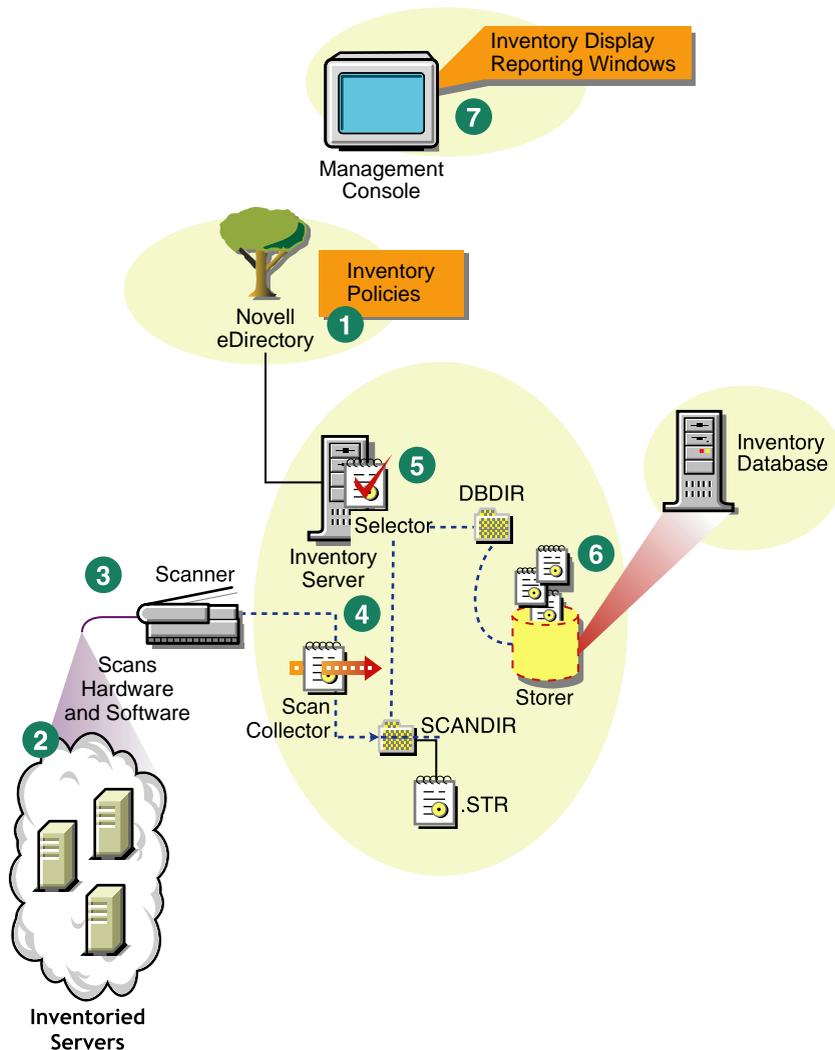
The Inventory database is a repository of inventory information of the inventoried servers. In ZfS, the database is a Common Information Model-based database but it is implemented in relational database management system (RDBMS) and maintained in Sybase* or Oracle*.

Management Console

The management console uses ConsoleOne, the Novell single management tool for administration. This is a Java-based console that includes snap-ins for Server Inventory management operations.

Understanding the Inventory Scanning Cycle in the Standalone Scenario

The following illustration depicts the scanning components and the inventory scanning cycle in the standalone scenario, which is explained below:



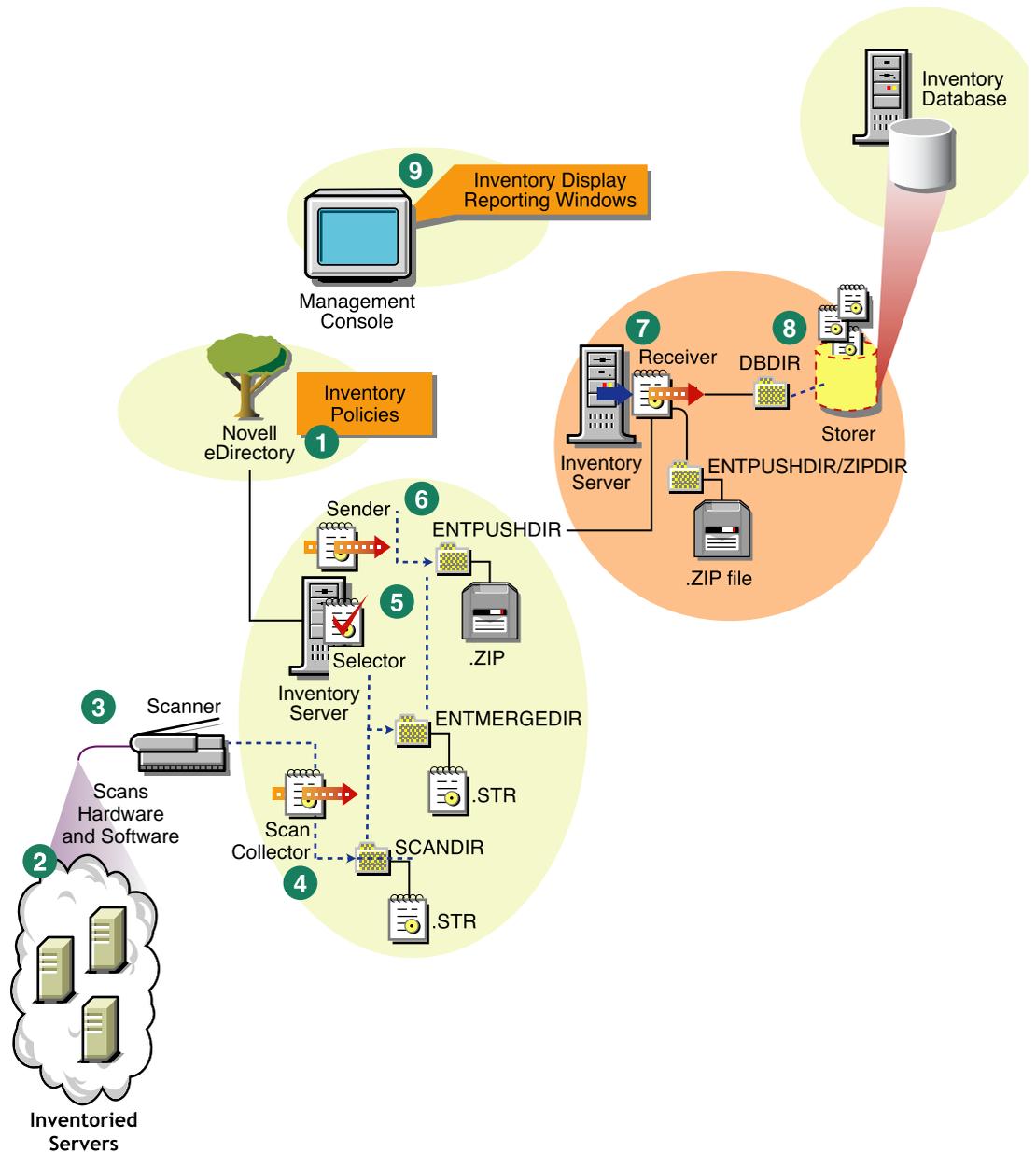
The inventory scanning cycle is as follows:

1. The inventory policies in the eDirectory define the inventory settings, such as the Inventory Service object name of the Inventory server to which the scan data will be sent, scanning time, whether to include software scanning of inventoried servers, and the software rules for software scan. These settings are customizable.
2. The Scanner uses Policy and Distribution Services to read the inventory policies and collects the inventory information based on the policy settings.
3. The Scanner stores the scan data (.str) locally on the inventoried server. This data is transferred to the Inventory server using the XML-RPC protocol.
4. The Scan Collector receives the .str file using the XML-RPC protocol and stores the str file in the scan directory (scandir) at the Inventory server. The Scan Collector uses the ZEN Web Server to process the XML-RPC requests.

5. The Selector validates the .str file and places the file in the Database directory (dbdir).
6. The Storer updates the database with the inventory information of the .str file.
7. The network administrator views the inventory information and queries the database in ConsoleOne.

Understanding Rolling Up Scan Data Across Servers

The following illustration depicts rolling up the scan data across servers, which is explained below:



If the inventory deployment rolls up scan data across servers, the process of scanning is as follows:

1. The inventory policies in eDirectory define the inventory settings, such as the Inventory Service object name of the Inventory server to which the scan data will be sent, scanning time, whether to include software scanning of inventoried servers, and the software rules for software scan. These settings are customizable.
2. The Scanner uses Policy and Distribution Services to read the inventory policies and collects the inventory information based on the policy settings.
3. The Scanner stores the scanned data (.str) locally on the inventoried server. This data is transferred to the Inventory server using the XML-RPC protocol.
4. The Scan Collector receives the .str file using the XML-RPC protocol and stores the .str file in the scan directory (scandir) at the Inventory server. The Scan Collector uses the ZEN Web Server to process the XML-RPC requests.
5. The Selector validates the .str file and places the file in the enterprise merge directory (entmergedir) for roll-up of scan data. If there is a database attached, the Selector also places the files in the database directory (dbdir).
6. The Sender on the Inventory server has a Roll-Up policy to identify the Inventory server to which it will transmit the scan data and the Roll-Up schedule specifies the time for roll-up of data. The Sender compresses the .str files as a .zip file and places the .zip file in the enterprise push directory (entpushdir). The Sender then sends the .zip file to the Receiver on the next-level Inventory server.
7. The Receiver on the next-level Inventory server receives the .zip file.

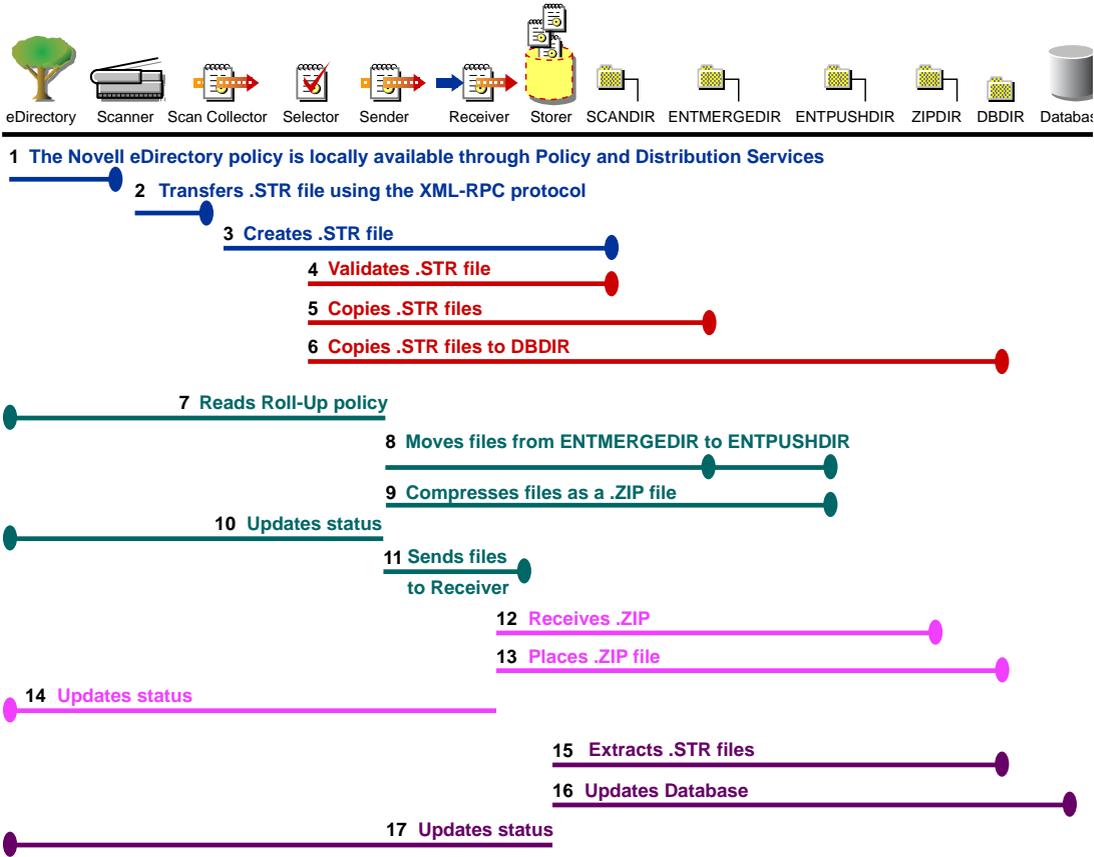
NOTE: The next-level Inventory server can be located on the same eDirectory tree or on a different eDirectory tree.

On the Intermediate Server, the Receiver copies the file in the enterprise push directory (entpushdir). On the Intermediate Server with Database, or the Intermediate Server with Database and Inventoried Servers, the Receiver places the file in entpushdir and places the file to the database directory (dbdir).

On the Root Server, or the Root Server with Inventoried Servers, the Receiver copies the file to the dbdir directory only.

8. The Storer extracts the .zip file containing the .str files to a temp directory (dbdir\temp) and updates the database with the inventory information of the inventoried server .str file.
9. The network administrator views the inventory information, and queries the database in ConsoleOne.

The following illustration lists the sequence of scan operations done by each Inventory component:



26

Setting Up Server Inventory

Before you install Novell® ZENworks® for Servers (ZfS) Server Inventory in your working environment, you must plan and decide the Inventory server tree hierarchy for your company. You should organize your inventory deployment based on your network and information requirements.

The following sections contain detailed information to help you deploy Server Inventory in your enterprise:

1. [“Understanding the Inventory Server Roles” on page 661](#)
2. [“Deploying Server Inventory” on page 670](#)
3. [“Understanding the Effects of Server Inventory Installation” on page 683](#)
4. [“Setting Up the Inventory Database” on page 684](#)
5. [“Configuring Inventory Servers for Server Inventory” on page 705](#)
6. [“Starting and Stopping the Inventory Service” on page 712](#)

You can change to role of the Inventory server. For more information, see [“Changing the Role of the Inventory Server” on page 713](#).

Understanding the Inventory Server Roles

This section describes the following roles that you assign for an Inventory server:

- ◆ [“Root Server” on page 661](#)
- ◆ [“Root Server with Inventoried Servers” on page 663](#)
- ◆ [“Leaf Server” on page 668](#)
- ◆ [“Leaf Server with Database” on page 669](#)
- ◆ [“Intermediate Server” on page 664](#)
- ◆ [“Intermediate Server with Database” on page 665](#)
- ◆ [“Intermediate Server with Database and Inventoried Servers” on page 667](#)
- ◆ [“Standalone Server” on page 670](#)

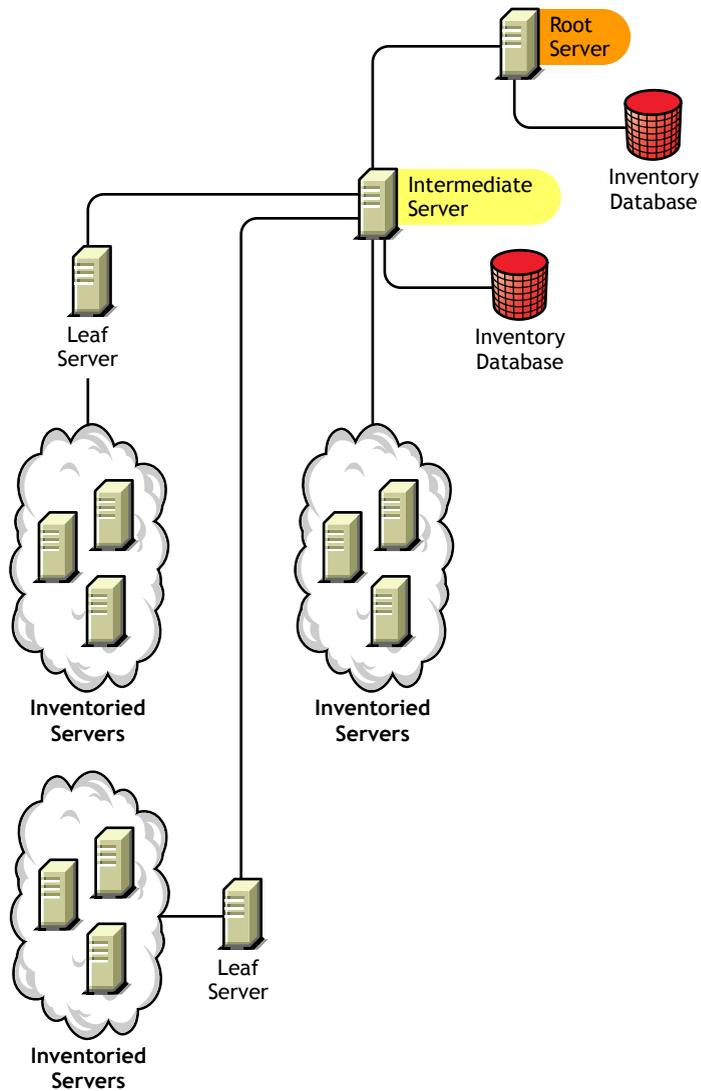
Root Server

The Root Server has the following characteristics:

- ◆ This server is the topmost Inventory server in the inventory tree hierarchy.
- ◆ This server has an Inventory database attached to it.

The Inventory database at the Root Server contains the inventory information for all the lower-level Inventory servers. At the Root Server level, you can view complete inventory information.

The following illustration depicts Leaf Servers connected to the Intermediate Server with Database. The Intermediate Server is attached to the Root Server.

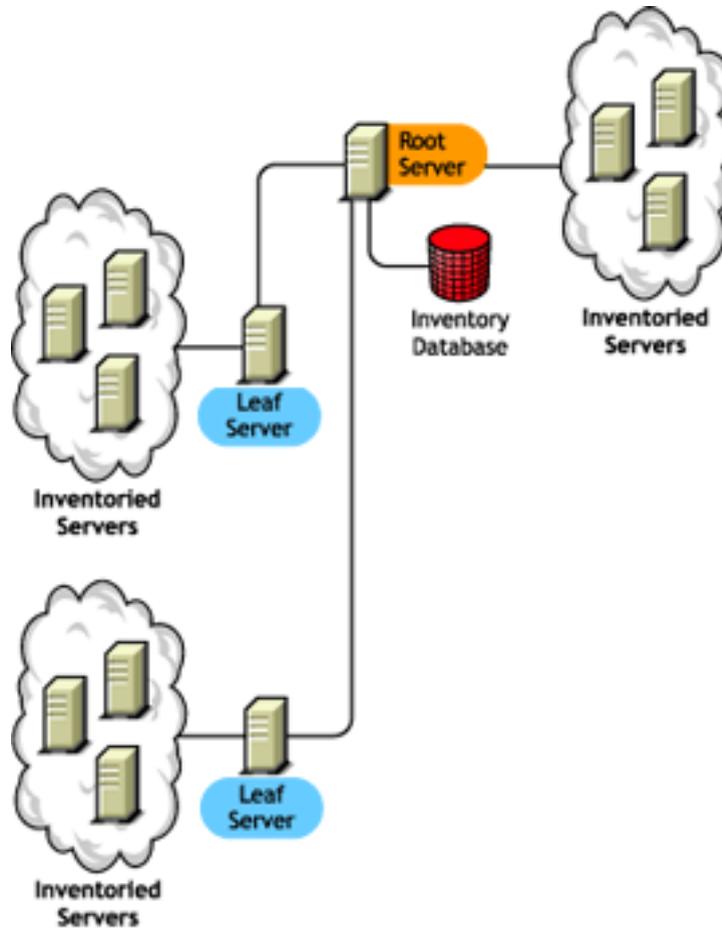


Root Server with Inventoried Servers

The Root Server with Inventoried Servers has the following characteristics:

- ◆ This server is the topmost Inventory server in the inventory tree hierarchy.
- ◆ This server has inventoried servers attached to it. There are inventoried servers residing on a LAN.
- ◆ This server has an Inventory database attached to it.

The following illustration depicts a Root Server with Inventoried Servers and Inventory database attached to it. The Leaf Servers are connected to the Root Server:

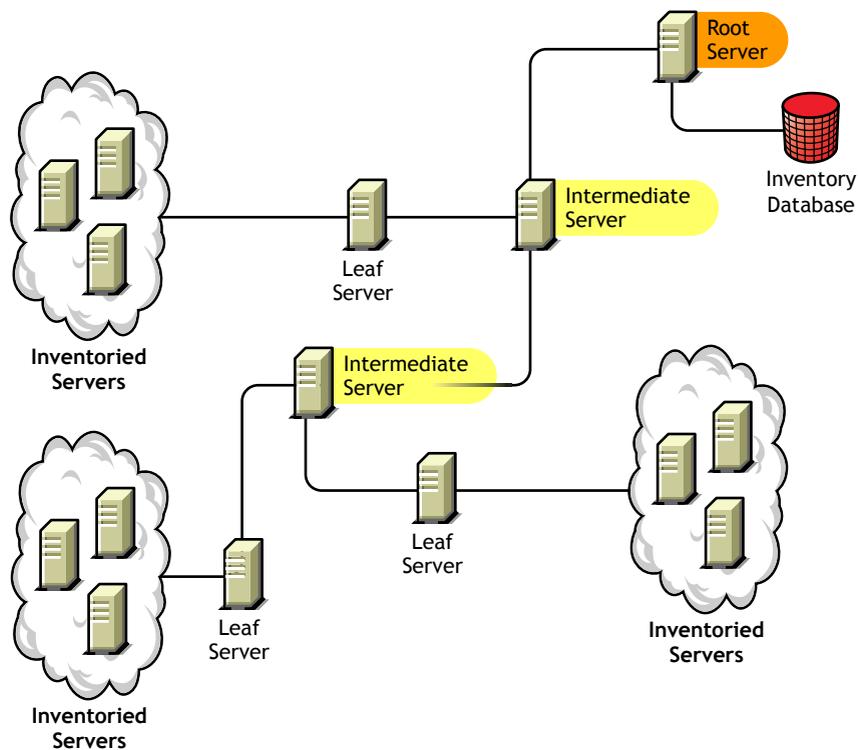


Intermediate Server

The Intermediate Server has the following characteristics:

- ◆ This Inventory server acts as a staging server for the lower-level Leaf Servers.
- ◆ This server moves the scan information to the next-level Inventory server or to the Root Server.
- ◆ This server does not have inventoried servers or an Inventory database attached to it.
- ◆ There can be one or more Intermediate Servers.

The following illustration depicts an Intermediate Server connected to Root Server. Two Leaf Servers roll up the inventory information to the Intermediate Server. This Intermediate Server rolls up the inventory information to another Intermediate Server that is connected to the Root Server.



There are many Leaf Servers and Intermediate servers at different levels. The Intermediate server is a staging server for uploading the scan information to the next-level server. The last Intermediate Server is attached to the topmost Root Server. This scenario is typical if there are many Leaf Servers in different geographical locations. All the Leaf Servers move the scan data to the Intermediate Server.

In some scenarios, the Leaf Server connects to the Intermediate Server over a WAN.

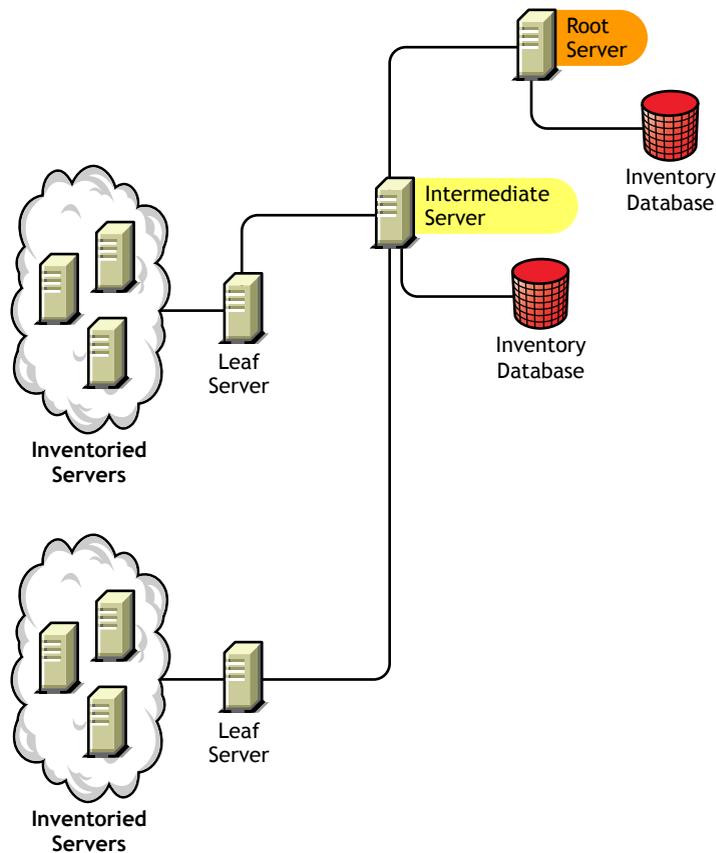
Intermediate Server with Database

The Intermediate Server with Database has the following characteristics:

- ◆ This server acts as a staging server for the lower-level Leaf Servers.
- ◆ This Inventory server moves the scan information to the next-level Intermediate Server or the Root Server.
- ◆ This server has an Inventory database attached to it.

There can be one or more Intermediate Servers in your enterprise.

The following illustration depicts two Leaf Servers attached to the Intermediate Server. A consolidated inventory information of all Leaf Servers is available at the Intermediate Server level.



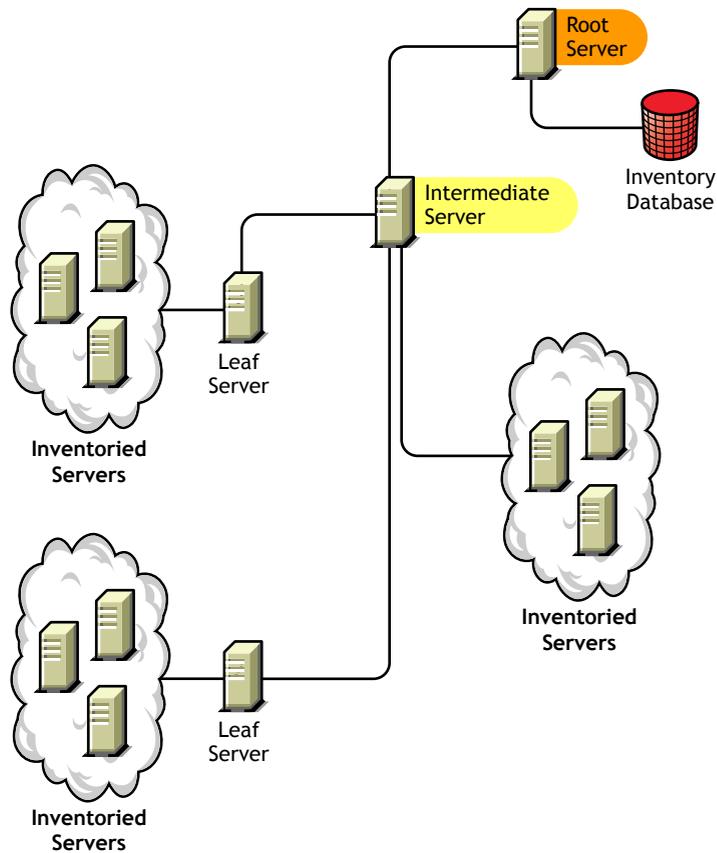
Intermediate Server with Inventoried Servers

The Intermediate Server with Inventoried Servers has the following characteristics:

- ◆ This Inventory server acts as an intermediate server for the lower-level Leaf Servers.
- ◆ This server moves the scan information to the next-level Intermediate Server or to the Root Server.
- ◆ This server has inventoried servers attached to it.
- ◆ This server does not have an Inventory database attached to it.

There can be one or more Intermediate Servers in your enterprise.

The following illustration depicts two Leaf Servers attached to the Intermediate Server. This Intermediate Server also has inventoried servers attached to it.

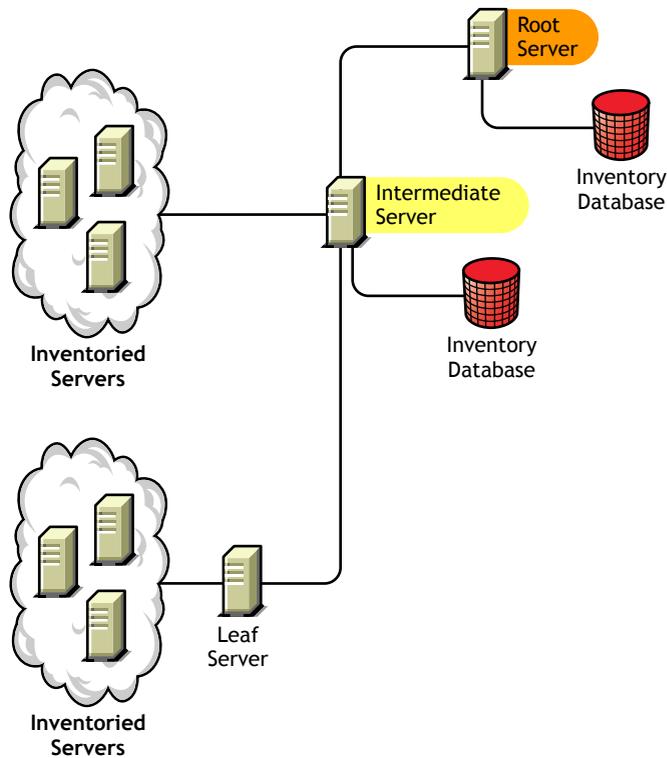


Intermediate Server with Database and Inventoried Servers

The Intermediate Server with Database and Inventoried Servers has the following characteristics:

- ◆ This Inventory server acts as an intermediate server for the lower-level Leaf Servers.
- ◆ This server moves the scan information to the next-level Intermediate Server or to the Root Server.
- ◆ This server has inventoried servers attached to it.
- ◆ This server has Inventory database attached to it.

The following illustration depicts two Leaf Servers attached to the Intermediate Server. The Intermediate Server has inventoried servers attached to it. A consolidated Inventory database of all Leaf Servers and the inventoried servers that are directly connected to the Intermediate Server is available at the Intermediate Server level.

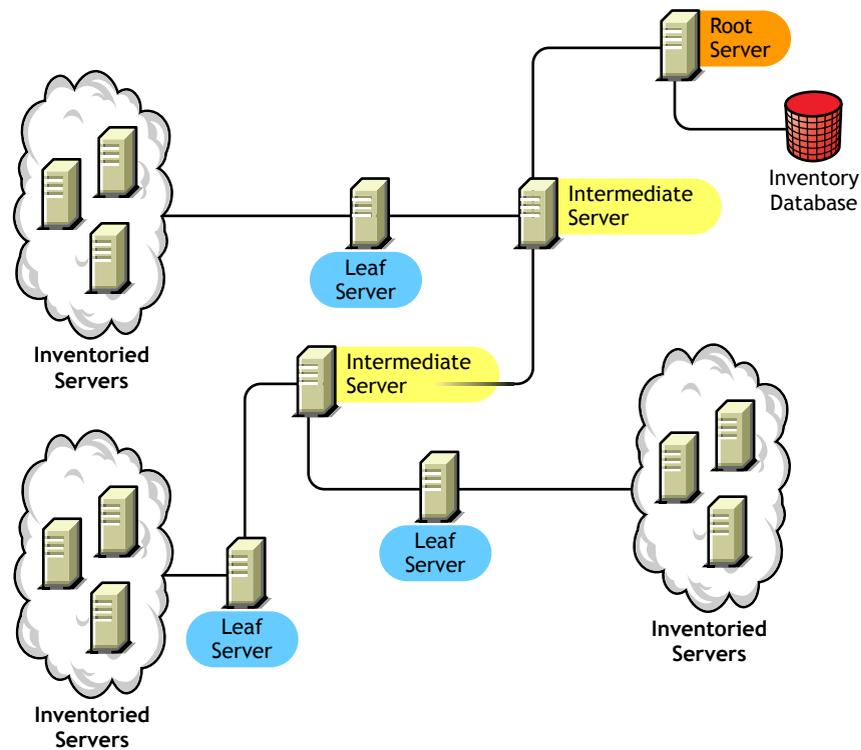


Leaf Server

The Leaf Server has the following characteristics:

- ◆ This Inventory server is at the lowest level in the hierarchy.
- ◆ This server has inventoried servers attached to it.
- ◆ This server moves the scan data to the next-level Intermediate Server or to a Root Server.
- ◆ A simple Leaf Server does not have an Inventory database. An Inventory database is not required because there may be only few inventoried servers connected to the Inventory server.

The following illustration depicts many Leaf Servers attached to the Intermediate Server. The Intermediate Server is connected to Root Server. A consolidated Inventory database of all Leaf Servers is available at the Root Server level.

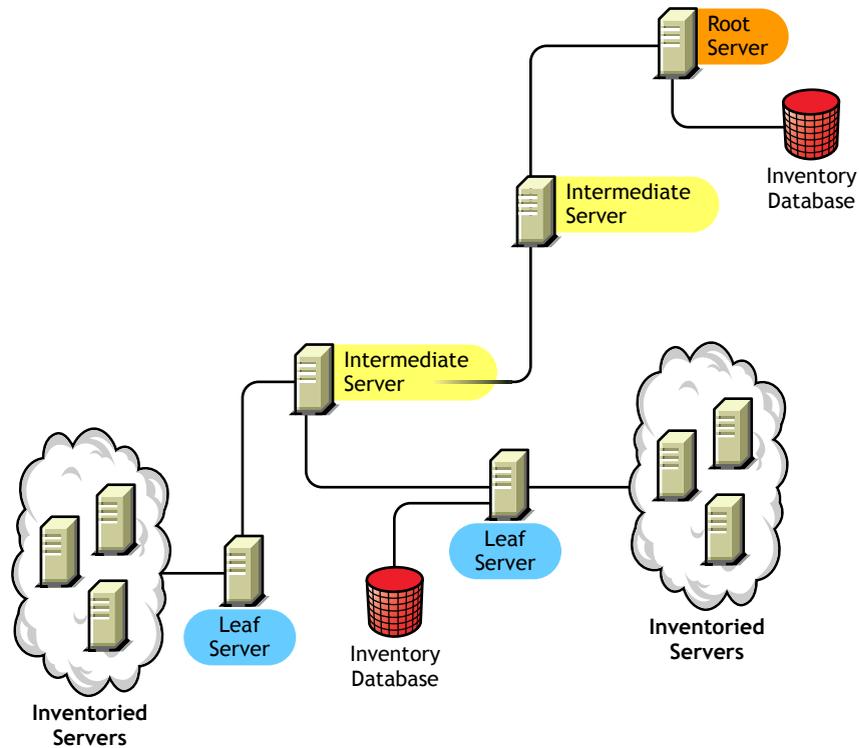


Leaf Server with Database

The Leaf Server with Database has the following characteristics:

- ◆ This Inventory server has inventoried servers attached to it.
- ◆ This server moves the scan data to the next-level Inventory server.
- ◆ This server has an Inventory database. You can assign a server as a Leaf Server with Database to maintain the inventory information for inventoried servers specific to the inventory site.

The following illustration depicts two Leaf Servers attached to the Intermediate Server. One Leaf Server has an Inventory database attached to it. This database contains a consolidated inventory of all inventoried servers attached to this Leaf Server.

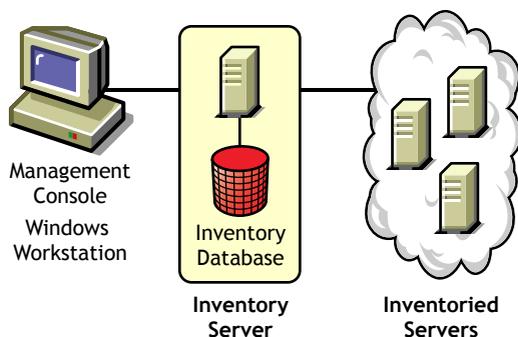


Standalone Server

The Standalone Server has the following characteristics:

- ◆ This server has inventoried servers attached to it.
- ◆ This server has an Inventory database attached to it.
- ◆ There is no roll-up of scan information and there are no requirements for Intermediate Servers and the Root Server.

The following illustration depicts Standalone Server.



Deploying Server Inventory

The following sections will help you to deploy Server Inventory:

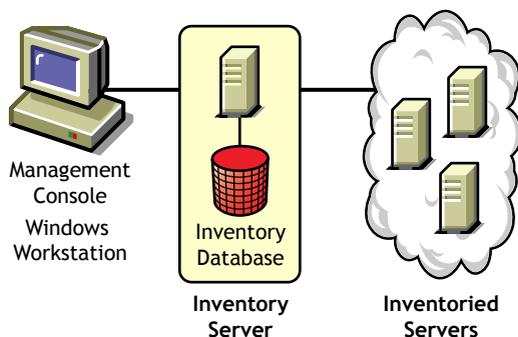
- ◆ [“Deploying Server Inventory in a LAN Environment” on page 670](#)
- ◆ [“Deploying Inventory over a WAN Environment” on page 671](#)

IMPORTANT: The recommendations discussed in the scenarios are generic. Because of the unique nature of your topology, further refinements may become necessary.

Deploying Server Inventory in a LAN Environment

In ZFS, the deployment of Server Inventory in a LAN environment implies deploying the product on a single inventory site.

In this type of inventory configuration, the Inventory server components and database are located on a Standalone Server. There is no roll-up of data and the Sender-Receiver components are not used. This scenario is illustrated in the following figure.



Recommendations for Deployment in a LAN Environment

- ◆ The minimum base Inventory server configuration includes 256 MB RAM and a database cache of 64 MB. For a higher inventoried server range, the Inventory server configuration is 512 MB RAM and a database cache of 128 MB.
- ◆ All inventoried servers should send the scan data to the nearest Inventory server on the LAN; policies must be created based on this information.
- ◆ The transmission of scan data from inventory servers can take several hours or even more than a day. The scanning is an ongoing background process.
- ◆ If many inventoried servers are attached to the same inventory server, we recommend that you do not schedule the scan of all inventoried servers at the same time, because this will stress the Novell eDirectory™ and the inventory server File System Services.
- ◆ Ensure that the time synchronization radius is set within 2 seconds.
- ◆ For all databases, the optimal database cache size requirement for the server may vary because of the server environment. Determine the database cache size that needs to be set by trying a range of cache sizes in the runtime environment. The default Sybase* database cache size is 32 MB.

Deploying Inventory over a WAN Environment

In a WAN environment, complete the following tasks, in order, to design the inventory tree and deploy inventory:

- ◆ “1. List the sites in the enterprise” on page 671
- ◆ “2. What is the ideal place for the Root Server?” on page 672
- ◆ “3. Is any other database needed?” on page 673
“Optional step: If another database is needed” on page 673
- ◆ “4. Identify the route for Inventory data” on page 673
- ◆ “5. Identify servers on each site for Inventory, Intermediate and Database Servers” on page 674
- ◆ “6. Identify the location of the Distributors” on page 675
- ◆ “7. Create the tree of servers for company Inventory collection” on page 675
- ◆ “8. Create an implementation plan” on page 675
- ◆ “9. Start the actual deployment” on page 675

“Guidelines for Sending Inventory Information in a WAN” on page 683 covers recommendations for deployment.

1. List the sites in the enterprise

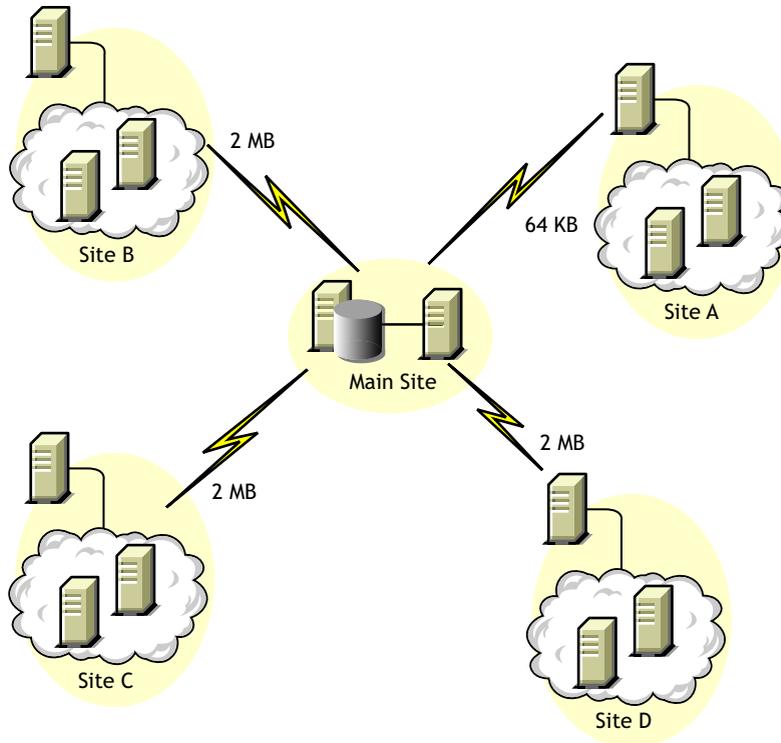
Describe the entire network of your company.

- ◆ List the various sites in your company.
- ◆ List the physical links between the various sites.
- ◆ Identify the type of links in terms of bandwidth and reliability.

The following figure illustrates the network organization of a company with servers in different locations.

Network Configuration of My Company

No. of NetWare Servers = 2
No. of Windows NT Servers = 5



This illustration depicts four sites (Site A, Site B, Site C, and Site D) connected to a central site. It depicts the physical links between the sites and the type of links in terms of bandwidth.

2. What is the ideal place for the Root Server?

The Root Server in the inventory tree is the highest-level server. Necessarily, an Inventory database is attached to the Root Server.

The inventory information available from the Inventory database of the Root Server will consist of all information from lower-level sites on the network and from the Root Server site.

Factors that you must consider include:

- ◆ There must be high-speed links between the Root Server and the management console.
- ◆ There must be high-speed links between the site having the Root Server and the sites having the lower-level Inventory servers.
- ◆ Using the management console, the administrator can collect the inventory information from any of the sites connected on high-speed links from the Root Server, or from the Root Server level site.
- ◆ A database server of suitable configuration should be provided for the Inventory server.

3. Is any other database needed?

Besides the database at the Root Server, you can maintain database servers at different sites.

You may want to maintain additional databases if there are sites or subtrees that are managed for inventory at different locations, and these sites are connected to the network over a slow link.

You should also determine if there are specific reasons to have a separate database for a single site or a set of sites. There may be some organizational needs for your company to have the database server on different sites, even if there is no product deployment need to have any other database.

NOTE: For a majority of enterprises, there may be no need to have any other database besides the enterprise-wide single database.

Optional step: If another database is needed

- ◆ If you decide to have additional database servers, identify the sites that need a database. Additionally, you need to examine whether the database will cater to the local site or a site with many subsites (subtrees). Also, identify the sites that require data in each Inventory database.
- ◆ All the sites served by a single database should typically access this database instead of the database at the Root Server for inventory management. This reduces the load on the database at Root Server.
- ◆ Database administrators should be available for these sites.

4. Identify the route for Inventory data

Identify the routes for inventory data for all sites to the nearest database, and then identify the route to the database on the Root Server.

To devise a route plan:

- ◆ Each route can have an Intermediate Server at a staging site. The Intermediate Server receives and transmits the data to the next destination. These are application-layer-level routes for inventory data. There can be various network-layer-level routes between two adjacent servers, which will be determined and managed by the routers in the network.
- ◆ The route provides information indicating how inventory data travels from a particular site to its final destination, which is the database at the Root Server.
- ◆ There may be multiple routes. Choose the fastest and most reliable route. To determine the route, consider the physical network links.
- ◆ Routes identified and made operational can be changed later, although there may be some cost in terms of management and traffic generation. If there is no intermediate database involved, you can change the route by only changing the eDirectory-based policy.
- ◆ Put Intermediate Servers on sites where the link parameters change substantially. Criteria to consider are difference in bandwidth, difference in reliability of the links, and the need for different scheduling.
- ◆ Availability of Inventory servers on the intermediate site for staging the inventory data should be considered in deciding the sites for Intermediate Servers. Provide enough disk space on these servers to store all the inventory data on the disk until the Sender sends it to the next destination.
- ◆ Inventoried servers should not be connected to the inventory server over a WAN because the inventoried server scanning should not be done across a WAN.

5. Identify servers on each site for Inventory, Intermediate and Database Servers

A single server can have different roles if it has sufficient resources. For example, an Inventory server can be a Leaf Server with Database. You can also designate an Inventory server as an Intermediate Server with Database, which receives inventory from the inventoried servers and also has an Inventory database. An Inventory server can have any combination of roles.

In ZfS, you choose the role for each Inventory server. For more information, see [“Understanding the Inventory Server Roles” on page 661](#).

The number of inventoried servers attached to an Inventory server also determines the load. The following table lists the disk space requirements for the server:

Server Type	Disk Space Requirements
Leaf Server	$(n1 \times s) + (n1 \times z)$
Leaf Server with Database	$(n1 \times s \times 2) + \{(n1 \times dbg)\}$
Intermediate Server	$n2 \times z$
Intermediate Server with Database	$(n2 \times z) + (n2 \times s) + \{(n2 \times dbg)\}$
Intermediate Server with Inventoried Servers	$(n1 \times s \times 2) + (n2 \times z)$
Intermediate Server with Database and Inventoried Servers	$(n1 \times s \times 2) + (n2 \times z) + (n2 \times s) + \{(n1 \times dbg) + (n2 \times dbg)\}$
Root Server	$(n2 \times z) + (n2 \times s) + \{(n2 \times dbg)\}$
Root Server with Inventoried Servers	$(n1 \times s \times 2) + (n2 \times z) + (n2 \times s) + \{(n1 \times dbg) + (n2 \times dbg)\}$
Standalone Server	$(n1 \times s \times 1) + \{(n1 \times dbg)\}$

In the table, $n1$ is the number of inventoried servers attached to the server.

s is the size of the scan data files. This file size varies depending on the data collected. Calculate 50 to 60 KB scan data from each inventoried server to calculate the load.

dbg is the storage space of the scan data in the database. Calculate 100 to 120 KB per inventoried server as the disk space for the database.

$n2$ is the number of inventoried servers rolled up to the Inventory server.

z is the size of the compressed scan data file per inventoried server. Calculate 7 to 10 KB for the roll-up of 50 KB scan data.

{ } denotes the disk space of the database server, depending on whether the database is on the same Inventory server or if it is connected to the Inventory server. If the database is on the same Inventory server, calculate the total disk space including the database space for the Inventory server. For example, if the Leaf Server with Database has the Inventory database on the same server, calculate the requirements for storage of scan data, including the database disk space.

6. Identify the location of the Distributors

The ZfS 3 Distributor component is required to distribute the inventory policies among the inventoried servers. For more information, see [Chapter 14, “Configuring Policy and Distribution Services,” on page 317](#).

7. Create the tree of servers for company Inventory collection

Ensure that the inventory tree you design follows these guidelines:

- ◆ The root of the tree is the Root Server.
- ◆ At least one Inventory server per site is recommended.
- ◆ Each site has inventoried servers to be scanned.
- ◆ Optionally, there will be databases and Intermediate Servers on different sites.

8. Create an implementation plan

After you design the inventory tree, you should develop an implementation plan to cover the phased deployment plan for the network. Use the top-down deployment of the Server Inventory installation. Always begin the installation at the topmost level server (Root Server) and proceed with the next lower-level servers.

9. Start the actual deployment

After your implementation plan is finalized, start the actual deployment according to the plan.

Follow these steps:

1. Install the Inventory servers on the sites.
2. Create the policies applicable to inventoried servers.
3. Create the Roll-Up policies to schedule the roll-up for each Inventory server.

Adding a Database Server to an Existing Inventory Setup

If you have already configured the servers for inventory setup, and you need to add another database server, follow these instructions:

- 1** Run the installation program to install the Inventory database on the server.
The installation program installs the Sybase database. If you are maintaining the database in Oracle*, make sure that the Oracle database exists. See [“Setting Up the Inventory Database for Oracle8i and Oracle9i” on page 690](#).
- 2** Shut down the Inventory services. For more information, see [“Stopping the Inventory Service” on page 713](#).
- 3** Based on the database you select, make sure that you configure the database. See [“Configuring the Database Location Policy” on page 710](#).

4 Modify the role of the existing Inventory server in the Inventory Service object.

If you are adding a new Inventory server, you need not modify the role of that server. If you want to change the role of the Inventory server, for example, from Leaf Server to Leaf Server with Database, you need to modify the role of the Inventory server in the Inventory Service object.

4a In ConsoleOne[®], right-click the Inventory Service object (*servername_ZenInvservice*), click Properties, then click the Inventory Service Object Properties tab.

4b Choose the new role of the Inventory Service object, then click Apply.

You will see a list of actions that you should follow based on the chosen role. For example, if you change the Root Server to Root Server with Inventoried Servers, you need to configure the Server Inventory policy for the inventoried servers that you have attached. Similarly, to change the role to any other Inventory server, you need to follow the instructions to make the role change effective.

Follow the actions that you need to change the role. For more information, see [“Changing the Role of the Inventory Server” on page 713](#).

5 Make sure that you enforce Full Scan for the Inventory Service object.

5a In ConsoleOne, right-click the Inventory Service object (*servername_ZenInvservice*), click Properties, then click the Inventory Service Object Properties tab.

5b Check the Enforce Full Scan option, then click OK.

6 Bring up the Inventory service.

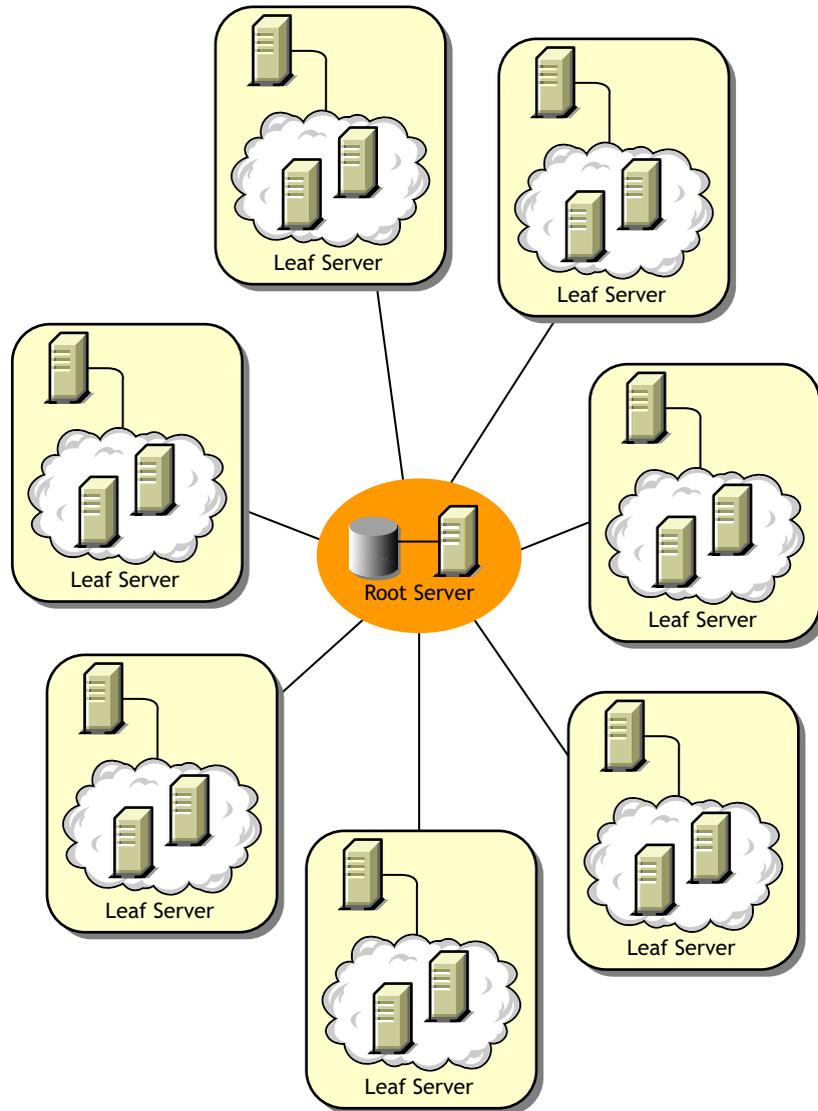
Possible Inventory Server Configurations for a WAN

The following sections cover these scenarios:

- ◆ [“Scenario 1: WAN Inventory Deployment for up to 50 Inventory Sites without Intermediate Servers” on page 677](#)
- ◆ [“Scenario 2: Up to 50 Intermediate Servers Connected to the Root Server” on page 678](#)
- ◆ [“Scenario 3: Intermediate Servers with Database Connected to the Root Server” on page 679](#)
- ◆ [“Scenario 4: Database on Inventory Servers and Intermediate Servers Connected to a Root Server” on page 680](#)
- ◆ [“Scenario 5: Roll-Up of the Inventory information Across eDirectory Trees” on page 681](#)
- ◆ [“Scenario 6: Merging eDirectory Trees” on page 682](#)
- ◆ [“Scenario 7: Deploying Inventory Server Across Firewall” on page 682](#)
- ◆ [“Guidelines for Sending Inventory Information in a WAN” on page 683](#)

Scenario 1: WAN Inventory Deployment for up to 50 Inventory Sites without Intermediate Servers

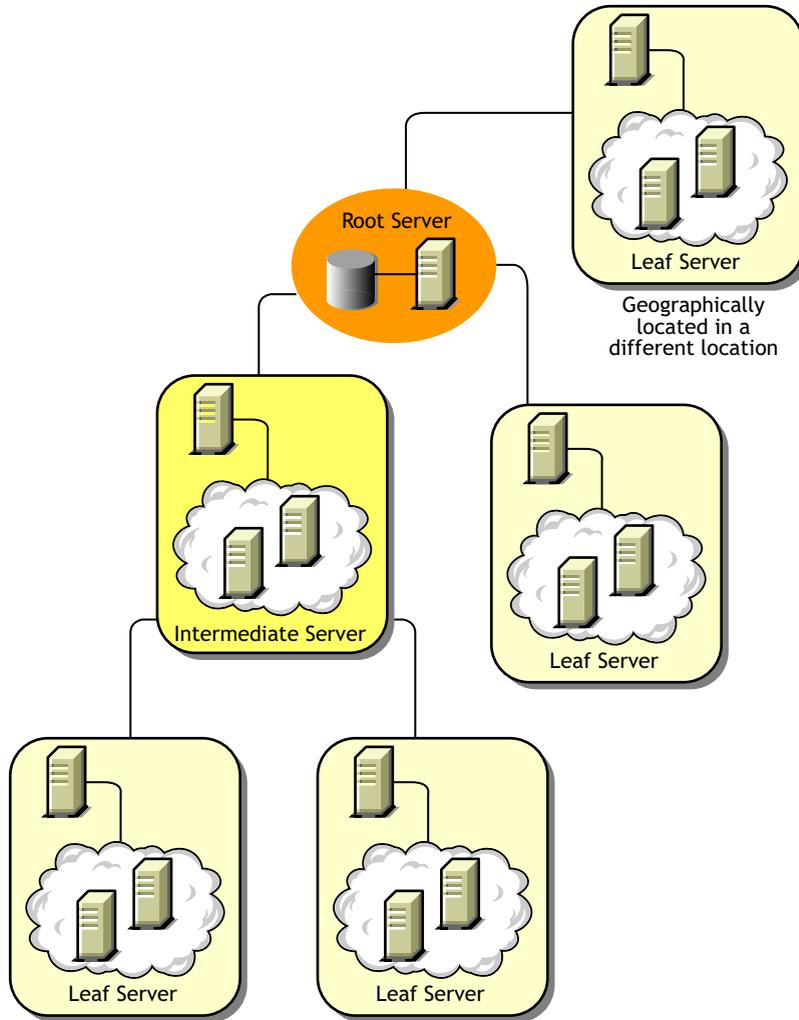
In this configuration, all Inventory servers are connected to a central enterprise database server. The Leaf Servers do not have a database and Intermediate Servers are not required. This scenario is illustrated in the following figure:



Scenario 2: Up to 50 Intermediate Servers Connected to the Root Server

In this configuration, the Leaf Servers roll up data to the next-level Intermediate Server and finally to the Root Server. Another Inventory server, at a different location, is also connected to the Root Server.

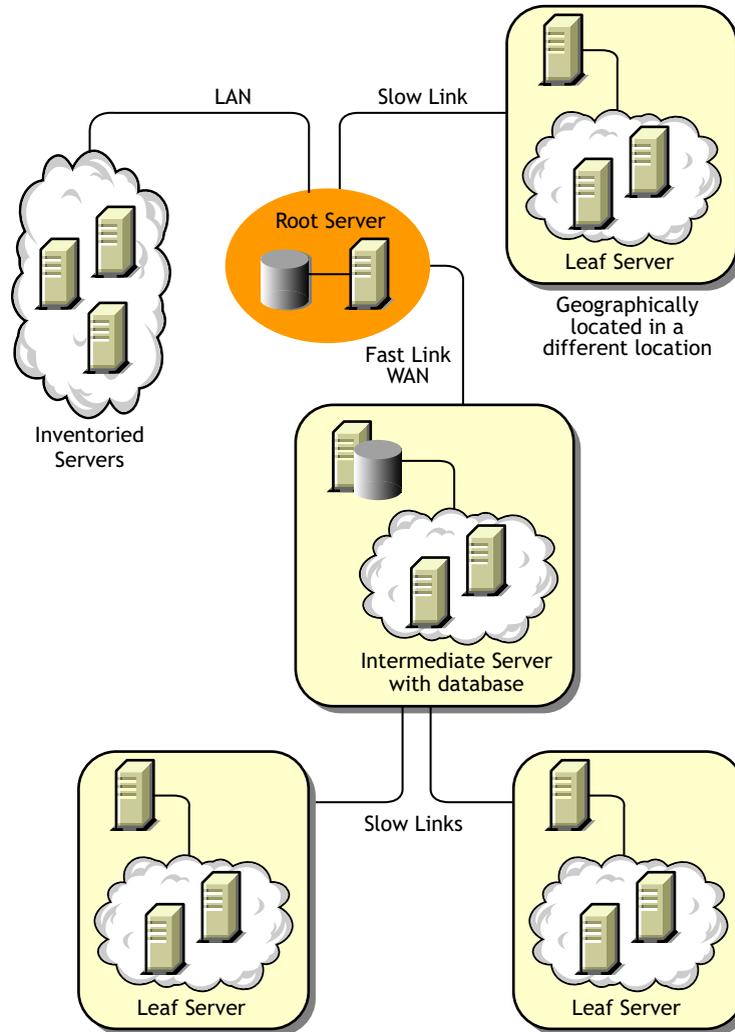
This scenario is illustrated in the following figure:



Scenario 3: Intermediate Servers with Database Connected to the Root Server

In this configuration, the inventory servers are connected to the Intermediate Server over fast WAN links. The Intermediate Server also has an Inventory database and transmits the information to the Root Server. Other Inventory servers are also connected to the Root Server.

This scenario is illustrated in the following figure:

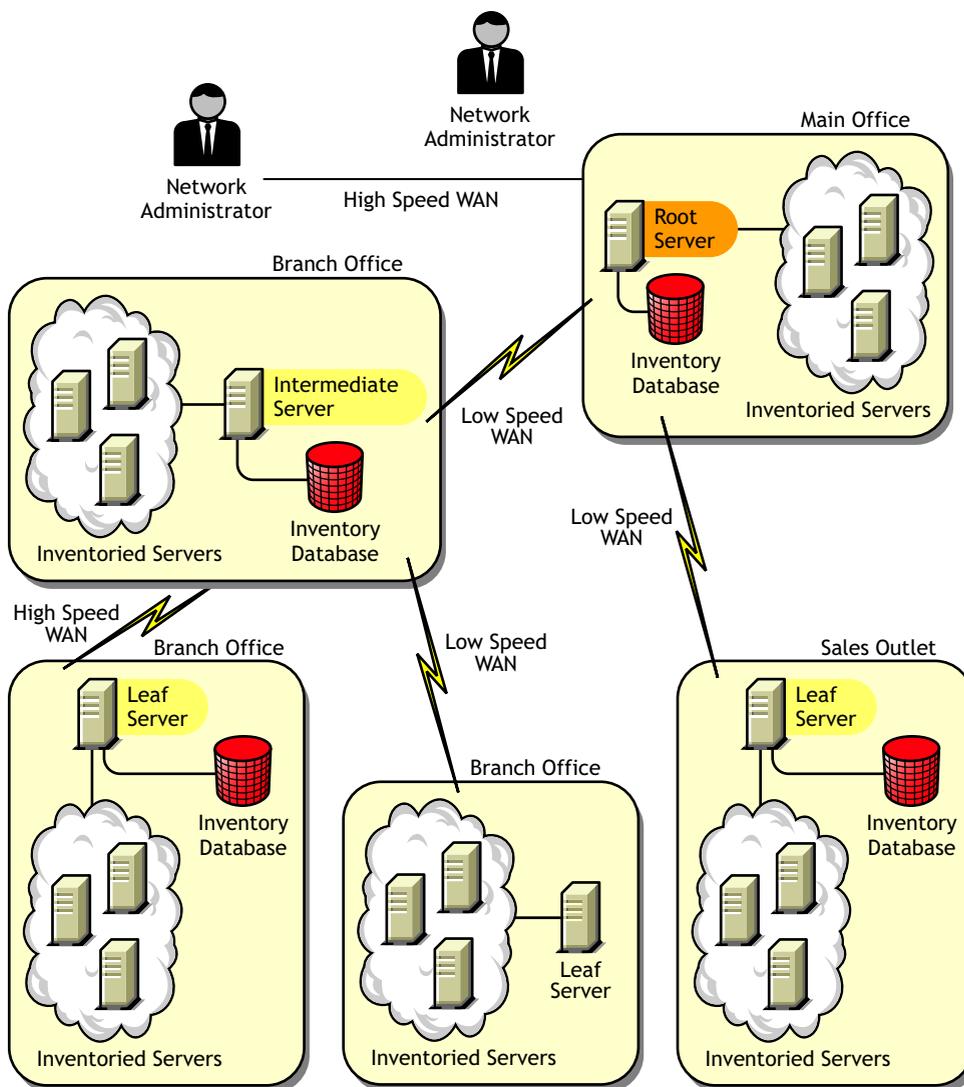


Scenario 4: Database on Inventory Servers and Intermediate Servers Connected to a Root Server

In this configuration, there are branch offices and a main office. Both branch offices store inventory information.

At one branch office, the Inventory server is a Leaf Server with Inventory Database, and the other branch office has a Leaf Server. At the next level, there is another branch office with an Intermediate Server with Database. The two branch offices at the lower level roll up data to this Intermediate Server. In turn, this Intermediate Server with Database rolls up data to the main office at the next level. There is also another sales outlet with a Leaf Server with Database at a sales outlet. This server directly rolls up data to the main office. The sales outlet and the two branch offices connect to the main office over low-speed WAN. One branch office connects to the main site over high-speed WAN.

This scenario is illustrated in the following figure:

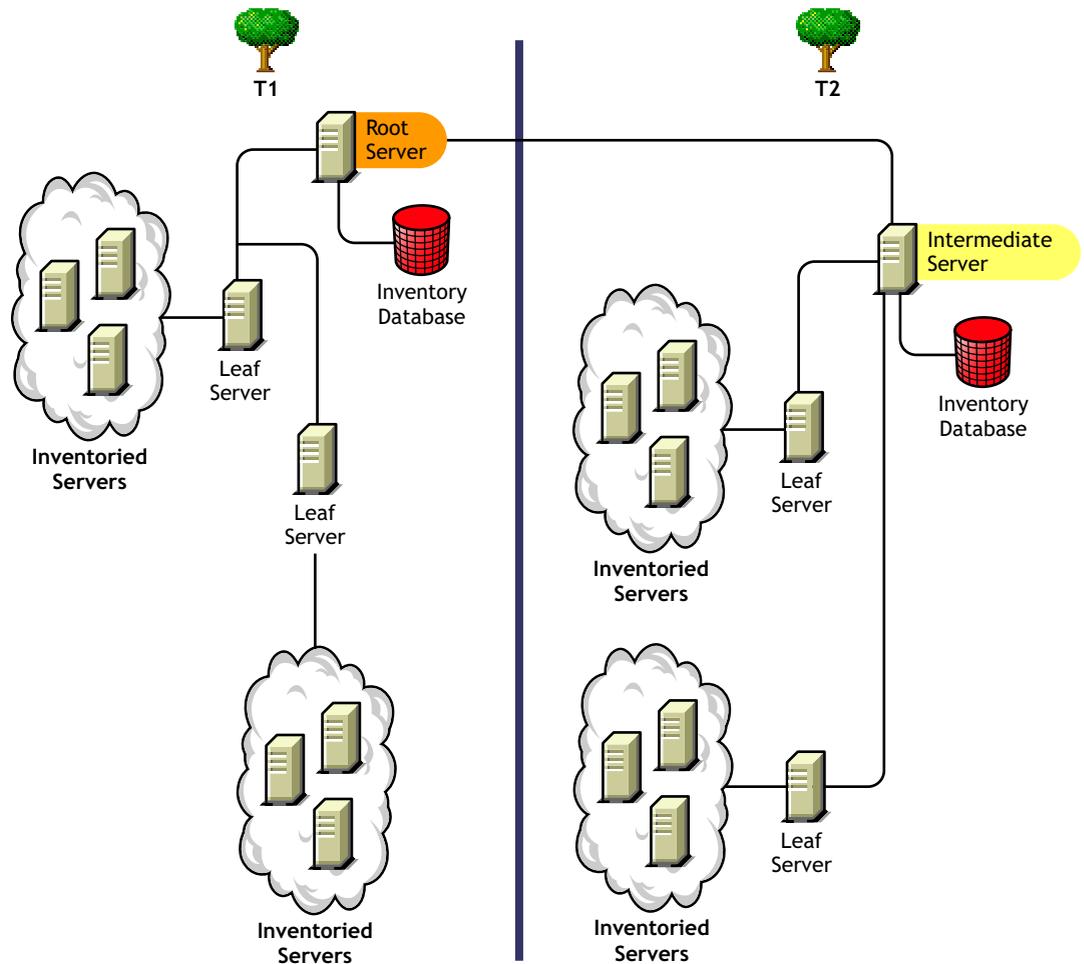


Scenario 5: Roll-Up of the Inventory information Across eDirectory Trees

In this configuration, you can deploy any of the previous scenarios. The highest-level Inventory server of one eDirectory tree rolls up the scan data to an Inventory server located on the other eDirectory tree.

In this configuration, you must install the Distributor on each eDirectory tree for the policies to be distributed.

The following illustration depicts a sample scenario where you can deploy this inventory configuration.



There are two organizations: A and B. Each organization has its own eDirectory tree and inventory tree. Organization A has two Leaf Servers and a Root Server in its inventory tree. Organization B also has two Leaf Servers and a Root Server in its inventory tree. A decision is taken to merge both the organizations and both the inventory trees but to retain the eDirectory trees. After the merger, the role of the Root Server on the eDirectory tree T2 is changed to Intermediate Server with Database and the scan data is rolled up from the Intermediate Server to the Root Server residing on the eDirectory tree T1.

Scenario 6: Merging eDirectory Trees

In this configuration, you can merge the inventory trees and the eDirectory trees. After you merge the eDirectory trees, you must manually change the eDirectory tree name and (optionally) the Inventory Service DN in the *Inventory_server_installation_directory\wminv\properties\config.properties* file before starting the Inventory service. For more information on merging the eDirectory trees, see the [Novell eDirectory documentation Web site \(http://www.novell.com/documentation\)](http://www.novell.com/documentation).

To merge the inventory trees, you must change the role of the Root Server of one inventory tree to roll up to an Inventory server in the other inventory tree.

To change the eDirectory tree name and the DN of an Inventory server, edit the following entries of the config.properties file:

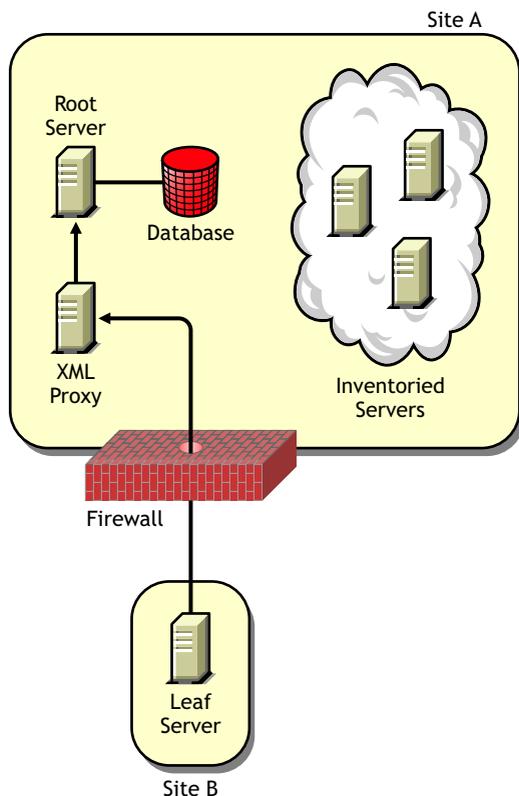
```
NDSTree=Target_eDirectory_tree_name
```

```
InventoryServiceDN=New_DN_of_the_Inventory_server
```

Scenario 7: Deploying Inventory Server Across Firewall

There are two sites; Site A and Site B connected through a WAN link. The Inventory server of Site A rolls up to an Inventory server in Site B. All communication from Site A to Site B flows through the firewall at Site B.

The following illustration depicts a sample scenario where you can deploy this inventory configuration:



Guidelines for Sending Inventory Information in a WAN

In this type of inventory deployment, the scanners transmit information to the servers over a WAN or dial-up connection.

- ◆ When you configure the inventory scanning of inventoried servers, we recommend staggering the inventory scanning to scan at different times or to scan some inventoried servers at a time.
- ◆ If many inventoried servers are attached to the same inventory server, we recommend that you do not schedule the scan of all inventoried servers at the same time, because this will stress the Novell eDirectory and the inventory server File System Service.
- ◆ You can attach inventoried servers to the server as determined by the number of connections supported by NetWare® or Windows* NT*/2000 servers up to a maximum of 5,000 inventoried servers.
- ◆ When you schedule the roll-up of data in the Inventory policies, we recommend the roll-up frequency should be at least one day. If the roll-up of scan data is scheduled too frequently, for example less than one hour, there may be some performance degradation of the inventory server.

Understanding the Effects of Server Inventory Installation

On the Inventory server, the ZfS 3 Server Inventory installation program does the following:

- ◆ On a NetWare Inventory server:
 - ◆ Copies the inventory related files to the *installation_directory*.
 - ◆ Copies the Server Inventory snap-in component to the ConsoleOne directory.
 - ◆ Creates an Inventory Service object (*servername_ZenInvservice*) in eDirectory for each server on which the Inventory server is installed. This object is populated with the following attributes: *zeninvRole* (role of the server), *zeninvScanFilePath* (path to the *scandir* directory), and *zeninvHostServer* (DN of the server on which Inventory server is installed).
 - ◆ If the Inventory Service object already exists, the object is validated and re-created if it is invalid.
 - ◆ During installation, the Inventory Service object is made a trustee of the NCP™ server with compare and read rights.
 - ◆ The installation program assigns the Inventory Service object as trustee to itself.
 - ◆ Creates the scan directory (*scandir*) with the subdirectories (*entpushdir*, *entmerge*, and *dbdir*) in the specified volume on the Inventory server.
 - ◆ Creates the *zenworks.properties* file in *sys:\system*. This file contains the installation path of the Inventory server and the ZEN Web server.
 - ◆ Installs the ZEN Web server on the Inventory server, if not installed previously.
 - ◆ If Server Inventory is reinstalled in the same directory as the previous installation, the *config.properties* and *directory.properties* files are backed up and re-created.
- ◆ On a Windows NT/2000 Inventory server:
 - ◆ Copies the inventory related files to the *installation_directory*.
 - ◆ Copies the Server Inventory snap-in component to the ConsoleOne directory.
 - ◆ Creates the *scandir* directory with the subdirectories.

- ◆ Creates an Inventory Service object (*servername_ZenInvservice*) in eDirectory for each server on which the Inventory server is installed. The following attributes are populated: *zeninvRole* (Role of the server), *zeninvScanFilePath* (Path to scandir), and *zeninvHostServer* (DN of the server on which Inventory is installed).
- ◆ The installation program assigns the Inventory Service object as trustee to itself.
- ◆ On the Inventory server, the Inventory Service Manager is created as a service.
- ◆ Edits the Registry settings to add the installation path of the Inventory server and the ZEN Web server.
- ◆ On the Inventory server, the ZEN Web server is created as a service.
- ◆ If Server Inventory is reinstalled in the same directory as the previous installation directory, the *config.properties* and *directory.properties* files are backed up and re-created.

On the Database server, the Server Inventory installation program does the following:

- ◆ Installs the Sybase ASA 7.0.2.1583 (on NetWare) or Sybase ASA 7.0.2.1540 (on Windows NT/2000) and the Inventory database on the servers you specify.
- ◆ If the Database server is installed in the previous installation directory, the database files are re-created if they were found invalid or non-existing.
- ◆ If Sybase is already installed, only the database files are copied.
- ◆ On NetWare, the *mgmt.db* entries are added to the *sys:\system\mgmt.dbs.ncf* file. On Windows NT/2000, the *mgmt.db* entries are added to the registry.
- ◆ Creates a Database object (*servername_InventoryDatabase*) for Sybase and configures the properties of the object.

On an existing ZENworks for Servers 2 installation, the installation program performs the following tasks in addition to the tasks performed for a fresh installation:

- ◆ On a NetWare Inventory server, the ZfS 3 Server Inventory installation deletes the following:
 - ◆ *Inventory.ncf* from the *installation_directory\mwserver\bin* directory
 - ◆ The ZfS 2 Inventory entries (*gatherer.ncf*, *master.ncf*, and *storer.ncf*) from *sys:\system\autoexec.ncf*.
 - ◆ The *gpsv* and *storer* directories from *installation_directory\mwserver*.
- ◆ On the Database server, deletes *zeninv.db* from the list of databases that are loaded.

Setting Up the Inventory Database

The following sections contain detailed information to help you set up your Inventory database for Sybase and Oracle:

- ◆ [“Setting Up the Inventory Database for Sybase” on page 685](#)
- ◆ [“Setting Up the Inventory Database for Oracle8i and Oracle9i” on page 690](#)
- ◆ [“Setting Up the Inventory Database for MS SQL Server 2000” on page 701](#)

If you want to replace the Inventory database, always stop the Inventory services before replacing the database. Replace the database and restart the Inventory services. For more information, see [“Starting and Stopping the Inventory Service” on page 712](#).

Setting Up the Inventory Database for Sybase

This section contains the following information:

- ◆ “Manually Creating the Inventory Database Object for Sybase” on page 685
- ◆ “Organizing the Database Spaces for a Sybase Database on NetWare or Windows NT/2000 Servers (AlterDBSpace Tool)” on page 686
- ◆ “Understanding the Sybase Database Startup Parameters” on page 688
- ◆ “Optimizing the Performance of the Sybase Database” on page 688
- ◆ “Backing Up the Inventory Database Running Sybase” on page 689

Manually Creating the Inventory Database Object for Sybase

To manually create the Inventory database object for Sybase:

- 1** In ConsoleOne, right-click in the eDirectory tree where you want to create the database object, click New, click Object, click ZENworks Database, then click OK.
- 2** Enter a name for the database object, then click OK.
- 3** Configure the Database server options of the Database object.
 - 3a** In ConsoleOne, right-click the database object, click Properties, then click the ZENworks Database tab.
 - 3b** Select the database server object using any of the following methods:
 - ◆ If eDirectory is installed on the database server: in the Server DN field, browse for and select the Server object for the server where the database is physically installed and running.

The server's IP address is automatically populated to the Server IP Address or DNS Name drop-down list. If the selected server object has more than one IP address, select the appropriate IP address.
 - ◆ If eDirectory is not installed on the database server, then enter the server's IP address or the DNS name in the Server IP Address or DNS Name field.

IMPORTANT: If the ZENworks database is located on a NetWare 4.x server, you must enter the server's IP address in the Server IP Address or DNS Name field instead of adding the server's object to the Server DN field.
 - 3c** Type the values for the following options:
 - ◆ **Database (Read-Write) Username:** *MW_DBA*
 - ◆ **Database (Read-Write) Password:** *novell*
 - ◆ **Database (Read Only) Username:** *MW_READER*
 - ◆ **Database (Read Only) Password:** *novell*
 - ◆ **Database (Write Only) Username:** *MW_UPDATER*
 - ◆ **Database (Write Only) Password:** *novell*
 - 3d** Click Apply.
 - 3e** To configure the JDBC* Driver properties, click the Jdbc Driver Information tab.

3f Select Sybase, then click Default Settings.

This populates the fields with default JDBC driver information.

The database settings for Sybase are:

- ◆ **Driver:** *com.sybase.jdbc.SybDriver*
- ◆ **Protocol:** *jdbc:*
- ◆ **SubProtocol:** *sybase:*
- ◆ **SubName:** *Tds:*
- ◆ **Port:** *2638*
- ◆ **Flags:** *?ServiceName=mgmtdb&JCONNECT_VERSION=4*
- ◆ **Database Service Name:** *the database name specified against the -n Sybase startup parameter while invoking Sybase.*

NOTE: By default, the value of the `-n` switch is the IP address of the database server. If you retain this switch value, you must enter the same IP address as the database service name.

3g Click Apply, then click Close.

Organizing the Database Spaces for a Sybase Database on NetWare or Windows NT/2000 Servers (AlterDBSpace Tool)

If a NetWare database server has volumes other than SYS: or a Windows database server has additional hard drives, placing the Sybase database spaces files on separate volumes or drives improves performance while accessing the database.

If you install the Sybase database component of ZFS 3, the system database file and the database spaces files are installed in the location on the database server you specify. On loading the Inventory database server, the system database file (`mgmtdb.db`) is loaded. This `mgmtdb.db` file references the inventory information in the database spaces files. The database spaces files (`mgmtdb1.db`, `mgmtdb2.db`, `mgmtdb3.db`, `mgmtdb4.db`, `mgmtdb5.db`, `mgmtdb6.db`, `mgmtdb7.db`, `mgmtdb8.db`, `mgmtdb9.db`, `mgmtdb10.db`, and `mgmtdb11.db`) contain the inventory information.

The `alterdb.props` file is installed on the database server in the `Inventory_server_installation_directory\wminv\properties` directory. You can modify the sections in the file to specify the location of the database spaces on the volumes or drives.

The contents of the `alterdb.props` file are as follows:

```
#Database Space Properties

count=11

mgmtdb1=location_of_mgmtdb1

mgmtdb2=location_of_mgmtdb2

mgmtdb3=location_of_mgmtdb3

mgmtdb4=location_of_mgmtdb4

mgmtdb5=location_of_mgmtdb5

mgmtdb6=location_of_mgmtdb6

mgmtdb7=location_of_mgmtdb7

mgmtdb8=location_of_mgmtdb8
```

```
mgmtdb9=location_of_mgmtdb9
mgmtdb10=location_of_mgmtdb10
mgmtdb11=location_of_mgmtdb11
.....
```

To organize the database spaces:

- 1** Ensure that the database is not loaded.
- 2** Ensure that the Inventory Service Manager is not running on the Inventory server.
- 3** Manually move the database spaces files on the Inventory server.

Arrange the database spaces files as follows for better performance:

- ◆ MGMTDB1 and MGMTDB2 in the same location
- ◆ MGMTDB3 and MGMTDB6 in the same location
- ◆ MGMTDB5 and MGMTDB7 in the same location
- ◆ MGMTDB8 and MGMTDB4 in the same location
- ◆ MGMTDB9 and MGMTDB10 in the same location
- ◆ MGMTDB11 in a location

IMPORTANT: If you move mgmtdb.db to another directory or volume on a NetWare server, update the sys:\system\mgmt dbs.ncf file with the new location of the mgmtdb.db.

If you move mgmtdb.db to another directory or volume on a Windows NT/2000 server, run the ntdbconfig.exe located in zenworks\dbengine directory. In the NTDBCONFIG dialog box, enter the new path of the mgmtdb.db.

- 4** Modify the location of the eleven database spaces files in the alterdb.props file.

For example, for NetWare, enter:

```
mgmtdb3=sys:\\zenworks\\inv\\db
```

or for windows nt/2000, enter:

```
mgmtdb3=c:\\zenworks\\inv\\db
```

- 5** Load the database, then enter **mgmt dbs** on NetWare servers, or on Windows NT/2000 servers, run the database service.

Ignore the error messages displayed on the console. These messages are displayed because the database spaces files are not loaded.

- 6** Ensure that the Database Location policy has been configured.
- 7** On the Inventory server console, run the AlterDBSpace service, then enter **StartSer AlterDBSpace**.

On the Inventory server, the AlterDBSpace tool runs as a service.

You will see a message that the database is adjusted.

- 8** Exit the database and then load the database.

Ensure that there are no errors while loading the database. Errors indicate that the specified location of the database spaces files are incorrect or does not exist. Ensure that the path to the database spaces files is correct in the alterdb.props file and repeat the procedure to organize the database spaces files.

IMPORTANT: If you place the database spaces files in different volumes or drives, the log file should be placed in the same volume or drive as the System database file (mgmt.db).

Understanding the Sybase Database Startup Parameters

The startup parameters of the Sybase database are as follows:

- ◆ **-c:** Sets the initial memory reserves for caching database pages and other server information. For example, `-c 32M` reserves 32 MB cache size.
- ◆ **-gc:** Sets the maximum length of time in minutes that the database server runs without doing a checkpoint on each database. The default value is 60 minutes. For example, `-gc` sets the checkpoint time as 120 minutes.
- ◆ **-m:** Deletes the transaction log when a checkpoint is done, either at shutdown or as a result of a checkpoint scheduled by the server.
- ◆ **-n:** Specifies the host name of the database server. For example, `-n IP_address`.
- ◆ **-ti:** Disconnects the connections that have not submitted a request for a certain number of minutes. The default is 240 (4 hours). A client machine in the middle of the database transaction locks until the transaction ends or the connection terminates. The `-ti` option is provided to disconnect inactive connections and to free their locks. For example, specify `-ti 400`.
- ◆ **-x:** Specifies a communication link. For example, `-x tcpip` indicates a TCP/IP link.
- ◆ **database_installation_path:** Specifies the installation path of the Inventory database. For example, `c:\zenworks\inv\db\mgmt.db`.

Optimizing the Performance of the Sybase Database

Increasing the database cache size improves database performance.

You can improve the performance of the Inventory database maintained in Sybase on NetWare or Windows NT/2000 Inventory servers. The default database cache size is 32 MB; however, this database cache size may not be adequate for large databases.

You should change the database cache size to an optimum size. You must also consider server memory size while assigning a cache size. For example, if you have 128 MB RAM, then a cache size of 32 MB is recommended.

To change the database cache size on the NetWare database server:

- 1** Close all connections to the Inventory database.
- 2** Quit the Sybase server.
- 3** Open the `mgmt.dbs.ncf` file in the `sys:\system` directory.
- 4** Modify the `-c` parameter.
For example, `-c 64M` sets the cache size to 64 MB.
- 5** Save the file.
- 6** On the server console, load the Inventory database. Enter **MGMTDBS**.

To change the database cache size on a Windows NT/2000 database server:

- 1 Stop the Sybase service.

On Windows NT, in the Control Panel, double-click Services, select Novell Database - Sybase, then click Stop.

On Windows 2000, in the Control Panel, double-click Administrative Tools, double-click Services, select Novell Database - Sybase, then click Stop.

- 2 On the database server, run the ntdbconfig.exe file from the dbengine directory.

Ntdbconfig.exe is a ZENworks database configuration utility for the ZENworks database using Sybase on Windows NT/2000 servers. This utility enables you to reconfigure the Sybase service. For the list of parameters recommended by Sybase, see [“Understanding the Sybase Database Startup Parameters” on page 688](#).

- 3 Modify the -c parameter.

- 4 Click OK.

- 5 Restart the Sybase service.

On Windows NT, in the Control Panel, double-click Services, select Novell Database - Sybase, then click Start.

On Windows 2000, in the Control Panel, double-click Administrative Tools, double-click Services, select Novell Database - Sybase, then click Start.

Backing Up the Inventory Database Running Sybase

ZfS provides an option to back up the Inventory database running Sybase from the ConsoleOne and Inventory database running Oracle from the server. We recommend that you back up the database on a weekly basis. However, if you are tracking the inventory of servers frequently, increase the frequency of backup.

To back up the database on NetWare or Windows NT/2000 servers:

- 1 In ConsoleOne, click Tools, click ZENworks Inventory, then click Database Backup.

If you want to back up the latest information in the Inventory database, right-click the database object, click ZENworks Inventory, then click Database Backup.

- 2 Enter the path to the directory where the database backup will be saved.

If the Inventory database is running on a NetWare server, you can either enter the path or click Browse to browse for and select a directory. If you just enter the database backup directory name without specifying the complete path, the backup directory will be created in the SYS: directory.

If the Inventory database is running on a Windows machine, you must manually enter the backup directory path. If you just enter the database backup directory name without specifying the complete path, the backup directory will be created in the \winnt\system32 directory.

NOTE: If you want to back up the database to a non-existent directory, only one level of the new directory will be created. To back up the database to subdirectory, ensure that the primary directory already exists. For example, if you want to back up the database to a new c:\backup directory, the backup directory will be created and the database will be backed up. But if you want to back up the database to a new database directory, located under c:\backup, the backup directory must already exist.

3 Click Start Backup.

This backs up the database to the specified directory on the server running the database and overwrites any existing files without prompting about the overwrite.

To restore the database:

1 If the Inventory database server is up, stop the Storer service. At the database server console, enter **StopSer Storer**.

2 Exit the Sybase database.

On NetWare servers: At the database server prompt, enter **q** to stop the Sybase database.

On Windows NT, in the Control Panel, double-click Services, select Novell Database - Sybase, then click Stop.

On Windows 2000, in the Control Panel, double-click Administrative Tools, double-click Services, select Novell Database - Sybase, then click Stop.

3 Copy the backup files, overwriting the working database files.

4 Restart the database server.

The backup tool creates a log file, backupst.txt, located in the consoleone\consoleone_version\bin directory on NetWare and Windows NT/2000 servers. The log records the status of the backup operation. Open this text file to view the status of the backup. This file increases in size for every backup operation. Remove the existing contents of the file if you do not require the details.

Setting Up the Inventory Database for Oracle8i and Oracle9i

The following sections explain how to configure the Inventory database for Oracle8i and Oracle9i:

- ◆ [“Setting Up the Inventory Database for Oracle8i” on page 690](#)
- ◆ [“Setting Up the Inventory Database for Oracle9i” on page 693](#)
- ◆ [“Manually Creating the Inventory Database Object for Oracle” on page 695](#)
- ◆ [“Loading the Inventory Database as a Separate Oracle Instance” on page 696](#)
- ◆ [“Optimizing the Performance of the Oracle Database” on page 698](#)
- ◆ [“Backing Up the Inventory Database Running Oracle” on page 699](#)

Setting Up the Inventory Database for Oracle8i

- ◆ [“Creating the Inventory Database for Oracle8i on a NetWare Server” on page 690](#)
- ◆ [“Creating the Inventory Database for Oracle8i on UNIX” on page 691](#)
- ◆ [“Creating the Inventory Database for Oracle8i on a Windows NT/2000 Server” on page 692](#)

Creating the Inventory Database for Oracle8i on a NetWare Server

You must manually create the Inventory database for Oracle on NetWare servers.

Prerequisites for configuring the database include the following:

- Oracle8i (8.1.5.0.4) Enterprise Edition on NetWare must be installed on the server before configuring the Inventory database.
- To maintain the Inventory database in Oracle, Server Inventory requires that you have a minimum of twenty five Oracle user licenses.

- Oracle files should not be installed on an NFS-mounted volume on the file server.
- Oracle data files must reside on volumes that have block suballocation turned off.

Perform the following procedure to create the Inventory database on Oracle8i for NetWare:

- 1** Create a directory `sys:\schema` and copy all files from the following directories on the *ZENworks for Servers 3* product CD to the SCHEMA directory:
 - ♦ `zfs\rminv\database\oracle\common`
 - ♦ `zfs\rminv\oracle\netwarespecific`
- 2** Create the `user_specified_volumepath\zenworks\inventory\oracle\database\trace` directory structure. Here `user_specified_volumepath` refers to the user selected directory to create the database.
- 3** In `sys:\schema_create.sql`, replace all instances of **oracle:** with `user_specified_volumepath`.
- 4** In `sys:\schema\init.ora`, replace all instances of **oracle:** with `user_specified_volumepath`.
- 5** In `sys:\schema_start.sql`, replace all instances of **oracle:** with `user_specified_volumepath`.
- 6** Copy `sys:\schema\init.ora` to `user_specified_volumepath\zenworks\inventory\oracle\database`.
- 7** Copy `sys:\schema_start.sql` to `user_specified_volumepath\zenworks`.
- 8** At the command prompt, enter **ORALOAD** to start Oracle, if not started.
- 9** Ensure that no Oracle database is mounted.
- 10** At the command prompt, enter **svrmgr31** to load the Oracle Server Manager by
- 11** At the Oracle Server Manager prompt, enter **@sys:\schema\schemata.sql**.
Review the `sys:\schema\inv.log` file to ensure that the database has been created successfully. If the database has not been successfully created, `inv.log` will contain the one or more of the following error messages: Oracle not available, Out of space, Compilation error.
- 12** At the Oracle Server Manager prompt, enter **@<volumepath>\zenworks_start.sql** to start the Inventory database.

Creating the Inventory Database for Oracle8i on UNIX

Ensure that the following requirements are met:

- Oracle version
 - On Linux* 6.0 or above: Oracle 8.1.5, 8.1.6 or 8.1.7 Enterprise Edition
 - On Solaris* 6.2 or above on Sparc*/Intel*: Oracle 8.1.5, 8.1.6 or 8.1.7 Enterprise Edition
- System requirements
 - Hard disk free space: 700 MB or above
 - Primary memory: 512 MB or above
- To maintain the Inventory database in Oracle, Server Inventory requires that you have a minimum of twenty five Oracle user licenses.

You must manually create the Inventory database for Oracle8i on the UNIX* server by following the procedure below:

- 1** Log in as Oracle user.
- 2** Create a directory /schema and copy all files from the following directories on the *ZENworks for Servers 3* product CD to the schema directory:
 - ♦ zfs\rminv\database\oracle\common
 - ♦ zfs\rminv\oracle\unixspecific
- 3** Create the *user_specified_directory_path*/zenworks/inventory/oracle/database/trace directory structure.
- 4** In schema/init.ora, replace all instances of \$HOME with *user_specified_directory_path*.
- 5** In schema/_start.sql, replace all instances of \$HOME with *user_specified_directory_path*.
- 6** In schema/_create.sql, replace all instances of \$HOME with *user_specified_directory_path*.
- 7** In schema/_schema.sql, replace all instances of \$HOME with the schema directory created in Step 2.
- 8** Copy schema/init.ora to *user_specified_directory_path*/zenworks/inventory/oracle/database.
- 9** Copy schema/_start.sql to *user_specified_directory_path*/zenworks.
- 10** Ensure the Oracle services are up and running and no database is mounted.
- 11** At the command prompt, enter **svrmgr1** to load the Oracle Server Manager.
- 12** At the Oracle Server Manager prompt, enter **@\$HOME/schema/schema.sql**
Review the schema/inv.log file to ensure that the database has been created successfully. If the database has not been successfully created, schema/inv.log will contain the following error messages: Oracle not available, Out of space, Compilation error.
- 13** At the Oracle Server Manager prompt, enter **@user_specified_directory_path/zenworks/_start.sql** to start the Inventory database.

Creating the Inventory Database for Oracle8i on a Windows NT/2000 Server

You must manually create the Inventory database for Oracle on Windows NT/2000 servers.

Prerequisites for configuring the database include the following:

- Oracle 8.1.5, 8.1.6 or 8.1.7 Enterprise Edition must be installed on the server before configuring the Inventory database.
- To maintain the Inventory database in Oracle, Server Inventory requires that you have a minimum of twenty five Oracle user licenses.

Perform the following procedure to create the Inventory database on Oracle8i for Windows NT/2000:

- 1** Create a directory c:\schema and copy all files from the following directories on the *ZENworks for Servers 3* product CD to the schema directory:
 - ♦ zfs\rminv\database\oracle\common
 - ♦ zfs\rminv\oracle\winntspecific
- 2** Create the *user_specified_path*\zenworks\inventory\oracle\database\trace directory structure.
- 3** In c:\schema_create.sql, replace all instances of d: with *user_specified_path*.

- 4 In c:\schema\init.ora, replace all instances of d: with *user_specified_path*.
- 5 In c:\schema_start.sql, replace all instances of d: with *user_specified_path*.
- 6 Copy c:\schema\init.ora to *user_specified_path*\zenworks\inventory\oracle\database.
- 7 Copy c:\schema_start.sql to *user_specified_path*\zenworks.
- 8 Ensure that Oracle services are loaded correctly and no database is mounted.
- 9 Load the Oracle Server Manager by entering **within a dos box: svrmgr1**
- 10 At the Oracle Server Manager prompt, enter **@c:\schema\schema.sql**
Review the c:\schema\inv.log file to ensure that the database has been created successfully. If the database has not been successfully created, inv.log will contain the following error messages: Oracle not available, Out of space, Compilation error
- 11 At the Oracle Server Manager prompt, enter **@<path>\zenworks_start.sql** to start the Inventory database.

Setting Up the Inventory Database for Oracle9i

- ♦ “Creating the Inventory Database for Oracle9i on UNIX” on page 693
- ♦ “Creating the Inventory Database for Oracle9i on a Windows NT/2000/2003 Server” on page 694

Creating the Inventory Database for Oracle9i on UNIX

Ensure that the following requirements are met:

- Oracle9i release 2 must be installed on Linux or Solaris versions supported by Oracle9i
- System requirements
 - Hard disk free space: 2 GB or above
 - Primary memory: 512 MB or above
- To maintain the Inventory database in Oracle9i, Workstation Inventory requires that you have a minimum of 25 user licenses.
- ZENworks for Servers 3.0.2/SP 2 Interim Release 3 must be installed on the Inventory server. For more information about installing ZENworks for Servers 3.0.2/SP 2 Interim Release 3, see the *Readme of ZENworks for Servers 3.0.2/SP2 Interim Release 3* (<http://support.novell.com>).

You must manually create the Inventory database for Oracle9i on the UNIX server by following the procedure below:

- 1 Log in as an Oracle user.
- 2 Create a /schema directory and copy all files from the following directories to the schema directory:
 - ♦ *zfs302_ir3.exe_extracted_directory*\zenworks\products\rminv\database\oracle9i\common
 - ♦ *zfs302_ir3.exe_extracted_directory*\zenworks\products\rminv\database\oracle9i\unixspecific
- 3 Create the *user_specified_directory_path*\zenworks\inventory\oracle\database\trace directory structure.

- 4** In schema/init.ora, replace all instances of \$HOME with *user_specified_directory_path*.
- 5** In schema/_start.sql, replace all instances of \$HOME with *user_specified_directory_path*.
- 6** In schema/_create.sql, replace all instances of \$HOME with *user_specified_directory_path*.
- 7** In schema/schema.sql, replace all instances of \$HOME with the schema directory created in Step 2.
- 8** Copy schema/init.ora to *user_specified_directory_path:/zenworks/inventory/oracle/* database.
- 9** Copy schema/_start.sql to *user_specified_directory_path/zenworks*.
- 10** Ensure the Oracle services are up and running and no database is mounted.
- 11** At the command prompt, enter **sqlplus /nolog** to load the Oracle Server Manager.
- 12** At the Oracle Server Manager prompt (sqlplus prompt), enter **@\$HOME/schema/schema.sql**.

Review the schema/inv.log file to ensure that the database has been created successfully. If the database has not been successfully created, inv.log will contain the following error messages: Oracle not available, Out of space, Compilation error.
- 13** At the Oracle Server Manager prompt, enter **@user_specified_directory_path/zenworks/_start.sql** to start the Inventory database.

Creating the Inventory Database for Oracle9i on a Windows NT/2000/2003 Server

You must manually create the Inventory database for Oracle9i on Windows servers.

Prerequisites for configuring the database include the following:

- Oracle9i release 2 must be installed on the server before configuring the Inventory database.
- To maintain the Inventory database on Oracle, Workstation Inventory requires that you have a minimum of 25 user licenses.
- System requirements
Hard disk free space: 2 GB or above
Primary memory: 512 MB or above
- ZENworks for Servers 3.0.2/SP 2 Interim Release 3 must be installed on the Inventory server. For more information about installing ZENworks for Servers 3.0.2/SP 2 Interim Release 3, see the *Readme of ZENworks for Servers 3.0.2/SP2 Interim Release 3* (<http://support.novell.com>).

To create the Inventory database on Oracle9i for Windows:

- 1** Create a directory c:\schema and copy all files from the following directories to the schema directory:
 - ♦ *zfs302_ir3.exe_extracted_directory\zenworks\products\rminv\database\oracle9i*common
 - ♦ *zfs302_ir3.exe_extracted_directory\zenworks\products\rminv\database\oracle9i*winntspecific
- 2** Create the directory structure: *user_specified_path\zenworks\inventory\oracle*database\trace.
- 3** In c:\schema_create.sql, replace all instances of **d:** with *user_specified_path*.

- 4** In `c:\schema\init.ora`, replace all instances of **d:** with *user_specified_path*.
- 5** In `c:\schema_start.sql`, replace all instances of **d:** with *user_specified_path*.
If **d:** is not found, check and correct the path of `init.ora` in the database directory.
- 6** Copy `c:\schema\init.ora` to *user_specified_path\zenworks\inventory\oracle\database*.
- 7** Copy `c:\schema_start.sql` to *user_specified_path\zenworks*.
- 8** Ensure that Oracle services are loaded correctly and no database is mounted.
- 9** At the command prompt, enter `sqlplus /nolog` to load the Oracle server manager.
- 10** At the Oracle Server Manager prompt (sqlplus prompt), enter `@c:\schema\schema.sql`.

Review the `c:\schema\inv.log` file to ensure that the database has been created successfully. If the database has not been successfully created, `inv.log` will contain the following error messages: Oracle not available, Out of space, Compilation error.
- 11** At the Oracle Server Manager prompt, enter `@user_specified_path\zenworks_start.sql` to start the Inventory database.

Manually Creating the Inventory Database Object for Oracle

To manually create the Inventory database object for Oracle:

- 1** In ConsoleOne, right-click a location in the eDirectory tree for the database object, click New, click Object, click ZENworks Database, then click OK.
- 2** Type a name for the database object, then click OK.
- 3** Configure the database server options of the database object.
 - 3a** In ConsoleOne, right-click the database object, click Properties, then click the ZENworks Database tab.
 - 3b** Select the database server object using any of the following methods:
 - ◆ If eDirectory is installed on the database server: in the Server DN field, browse for and select the Server object of the server where the database is physically installed and running.

The server's IP address is automatically populated to the Server IP Address or DNS Name drop-down list. If the selected server object has more than one IP address, select the appropriate IP address.
 - ◆ If eDirectory is not installed on the database server, then enter the server's IP address or the DNS name in the Server IP Address or DNS Name field.

IMPORTANT: If the ZENworks database is located on a NetWare 4.x server, you must enter the server's IP address in the Server IP Address or DNS Name field instead of adding the server's object to the Server DN field.
 - 3c** Type the values for the following options:
 - ◆ **Database (Read-Write) Username:** *MW_DBA*
 - ◆ **Database (Read-Write) Password:** *novell*
 - ◆ **Database (Read Only) Username:** *MWO_READER*
 - ◆ **Database (Read Only) Password:** *novell*
 - ◆ **Database (Write Only) Username:** *MWO_UPDATER*

- ♦ **Database (Write Only) Password:** *novell*

3d Click Apply.

3e To configure the JDBC Driver properties, click the JDBC Driver Information tab.

3f Select Oracle, then click Default Settings.

This populates the fields with default JDBC driver information.

The database settings for Oracle are:

- ♦ **Driver:** *oracle.jdbc.driver.OracleDriver*
- ♦ **Protocol:** *jdbc:*
- ♦ **SubProtocol:** *oracle:*
- ♦ **SubName:** *thin:@*
- ♦ **Port:** *1521*
- ♦ **Flags:** Not applicable for Oracle
- ♦ **Database Service Name:** *orcl*. (The value for the SID is the same as assigned for the database instance.)

3g Click Apply, then click Close.

Loading the Inventory Database as a Separate Oracle Instance

The following sections explain the steps for configuring and running multiple Oracle database instances:

- ♦ [“Configuring and Running Multiple Oracle Database Instances on a NetWare Server” on page 696](#)
- ♦ [“Configuring and Running Multiple Oracle Database Instances on a Windows NT/2000 Server” on page 698](#)

Configuring and Running Multiple Oracle Database Instances on a NetWare Server

To configure and run multiple Oracle database instances:

- 1** Unload Oracle. At the database server prompt, enter **oraunld**.
- 2** Invoke the Net8 configuration utility. At the database server prompt, run **easycfg.ncf** to load the Net8 Easy configuration window.
- 3** Define a unique Oracle instance.
 - 3a** Click Config > Listener > Database > Add.
 - 3b** Assign values for Database Instance and Database Name in the Adding Instances Address window.

For example, assign Database Instance=*Indy* and Database Name=*mgmtdb*. In this configuration, the database instance is *zfs*. You can specify any database instance name. The Database Domain field should be left blank.
 - 3c** Click Accept, then click Save.

- 4** Configure the Listener for IPC. To run an Oracle system, the IPC and TCP addresses should be already be configured.
 - 4a** Click Config > Listener > Address. Ensure that IPC and TCP addresses are configured for the server.

The setting for IPC is *servername_LSNR*, and TCP is *IPaddress* or *hostname*. If these settings exist, click Cancel. Otherwise, assign the values for these settings > click Save.
- 5** Create an IPC alias.
 - 5a** Click Config > Database Alias. The window will list the aliases for IPC, SPX, TCP, and others. Click Add to add an alias name for the new instance.

Enter the following details:

 - ◆ **Database Alias:** *servername-databaseinstance-IPC*. For example, the database alias is *austr*, where *austr* is the server name, and *indy* is the database instance created earlier.
 - ◆ **Protocol:** *IPC*
 - ◆ **Service/Host Name or Key Name:** *server_name_LSNR*
 - ◆ **Database Instance:** *Indy*
 - 5b** Click Accept, then click Save.
 - 5c** To verify the configured alias name in the list window: Click Config > Database Alias, select the newly created alias, then click View.

View the properties of the database alias. Ensure that the properties are correct. If the property settings are incorrect, delete the alias (click Delete) and repeat Step 5.
- 6** Exit the EasyCfg tool. Click Config > Exit.
- 7** Create a password file for logging as *Internal* user for this instance. Enter **load orapwd81 file=oracle_volume:oracle_home\database\pwwdatabase_instance.ora password=password entries=2** where *oracle_volume* is the NetWare volume name of your Oracle installation, *PWDdatabase_instance.ORA* is the password filename, and *password* is any password that you specify.

For example, `load orapwd81 file=oracle:\orahome1\database\pwwindy.ora password=mgmtdb entries=2`. This password file will be created in the *oracle_volume:\DATABASE* directory. Ensure that the file exists in the directory.
- 8** Load the Oracle NLM™ software. At the database server prompt, enter **oraload**.
- 9** To set the newly created ZfS instance, load the Oracle Server Manager. At the database server prompt, enter **svrmgr31**.
- 10** Enter the following commands: **set instance servername-databaseinstance**. For example, `set instance austr-indy-ipc`. This displays that the newly created instance is started.
- 11** Enter **connect internal/password** where *password* is the password created in Step 7.
- 12** Mount the Inventory database.

- 13** Modify the `_start.sql` file located in `volumepath\zenworks`. Enter the following lines in the file:

```
set instance servername-databaseinstance-IPC  
  
shutdown normal
```

- 14** Create the Database object. In ConsoleOne, right-click a location in the tree for the Database object, click New, click Object, click ZENworks Database, then click OK.
- 15** Type a name for the Database object, then click OK
- 16** Configure the Database server options of the Database object. For more information, see [Step 3 on page 695](#) in “Manually Creating the Inventory Database Object for Oracle” on page 695

If you are loading multiple databases in separate Oracle instances, then each database reserves a separate Oracle SGA memory, where Oracle keeps all the database resources. In such environments, you should increase the amount of memory on the server. Refer to the documentation provided by Oracle.

Configuring and Running Multiple Oracle Database Instances on a Windows NT/2000 Server

To configure and run Oracle instances:

- 1** At the database server, run the Oracle Database Configuration Assistant. From the desktop Start menu, click Programs > Oracle > Database Administration > Oracle Database Configuration Assistant.
- 2** Click Create a Database > Next > Typical > Next > Copy Existing Database Files from the CD > Next.
- 3** Enter the following details:
 - ◆ **Global Database Alias:** `mgmtdb.your_windows_NT/2000_name`
 - ◆ **SID:** The value is automatically filled as `mgmtdb`.
- 4** Click Finish.

This allows for Oracle database creation. This process takes a significant amount of time. Ensure that the OracleServiceMGMTDB service is created and started.

- 5** Load the Inventory database.

Run the Oracle Server Manager. From the desktop menu, click Start > Run > SVRMGRL. Enter the following commands:

```
set instance mgmtdb  
  
connect internal/password_for_administrator
```

Optimizing the Performance of the Oracle Database

If you have an Inventory database on Oracle, you can improve the performance of the database when you generate the inventory reports or query the database.

You use the database buffer cache to store the most recently used data blocks. The database cache is determined as `DB_BLOCK_BUFFERS * DB_BLOCK_SIZE`. These parameters are specified in the `INIT.ORA` file in the `ZENWORKS\DATABASE` directory on the database server.

`DB_BLOCK_BUFFERS` specifies the number of database buffers. `DB_BLOCK_SIZE` specifies the size of each database buffer in bytes.

The size of each buffer in the buffer cache is equal to the size of the data block.

Oracle recommends that the database buffer cache for any Online Transaction Processing Application (OLTP) should have a hit ratio of about 90%, which is optimal.

The Zfs Inventory database on Oracle has an approximate 88% hit ratio with a database cache size of 24 MB for 128 MB RAM, which is about 20% of total memory.

If there is additional memory, you configure the database cache size by increasing the `DB_BLOCK_BUFFERS` parameter in the `init.ora` file.

Backing Up the Inventory Database Running Oracle

To back up the database running Oracle:

- 1** If the database server is up, stop the Storer service. At the database server console, enter **stopSer Storer**.
- 2** Load the Oracle Server Manager.
On NetWare server with Oracle, enter **svrmgr31**.
On Windows NT/2000 server with Oracle Enterprise Edition, from the taskbar, click Start > Run, enter **svrmgr1**.
- 3** Enter the following commands:
set instance databaservername-databaseinstance-IPC, where *databaseinstance* refers to the database instance that you have set up earlier. See [“Loading the Inventory Database as a Separate Oracle Instance” on page 696](#).
For example, **set instance austr-zfs3-ipc**.
- 4** Connect as an administrator. For example, if the administrator’s internal name is *internal*, at the Server Manager prompt, enter **connect internal/password**.
where *password* is the password created earlier. See [“Loading the Inventory Database as a Separate Oracle Instance” on page 696](#).
4a At the Server Manager prompt, enter **select name from v\$datafile;**
This displays the list of the data files that Server Inventory uses.
- 5** Ensure that no other databases are mounted. At the prompt, enter **shutdown normal**.
- 6** Disconnect and exit from the Server Manager. At the Server Manager prompt, enter **disconnect;**
Enter **exit;**
- 7** Copy the complete schema directory to a backup volume or disk.

After the backup is done, ensure that the backup copy of the database matches the original copy. Perform database verification to verify the integrity of the backup.

To verify the database integrity on a NetWare server with Oracle, enter **load DBV81.NLM FILE=path_to_the_database_file BLOCKSIZE=4096**

To verify the database integrity on a Windows NT/2000 server with Oracle, enter **DBV.EXE FILE=path_to_the_database_file BLOCKSIZE=4096**

Example: enter **DBV.EXE FILE=c:\schema\database\cim1.ora BLOCKSIZE=4096**

Also, run this command for the following files: cim1.ora, cim2.ora, cim3.ora, cim4.ora, cim5.ora, cim6.ora, cim7.ora, cim8.ora, cim9.ora, cim10.ora, cim11.ora, sys1.ora, and ctl1.ora.

If the database backup is successful, ensure that there are no error messages on the verified pages. Ensure that the following displayed parameters display a zero value: TOTAL PAGES FAILING (DATA)=0, TOTAL PAGES FAILING (INDEX)=0, and TOTAL PAGES MARKED CORRUPT=0.

To restore the database:

- 1** If the Inventory database server is up, stop the Storer service. At the database server console, enter **StopSer Storer**.
- 2** Load the Oracle Server Manager.
On a NetWare server with Oracle, enter **svrmgr31**.
On a Windows NT/2000 server with Oracle Enterprise Edition, from the taskbar, click Start > Run, enter **svrmgr1**.
- 3** Connect as an administrator. For example, if the administrator's internal name is *internal*, at the Server Manager prompt, enter **connect internal/
password_for_administrator**.
- 4** Ensure that no other databases are mounted. Enter **shutdown normal**.
- 5** Disconnect and exit from the Server Manager. At the Server Manager prompt, enter **disconnect;**
Enter **exit;**
- 6** Copy the database from the backup location.

If you copy the database to a different location than the earlier location, modify the location in the following files to specify the new path:

- ◆ Edit the init.ora file located in \zfd3\oracle\database to specify the new path for the following parameters:

```
control_files=location_of_ctl1.ora\ctl1.ora  
background_dump_dest=location_of_trace_dir\trace  
user_dump_dest=location_of_trace_dir\trace
```

- ◆ Edit the _start.sql file in the sys:\system to specify the location of init.ora file in the following parameter:

```
startup pfile=location_of_the_init.ora\init.ora
```

- ◆ Modify the location in the alterctrl.sql to specify new path.

For example, modify the existing data:\zfd3\oracle\database path to oracle:\zfd3\oracle\database in alterctrl.sql.

In this .SQL file, modify the path for the following parameters, if required.

```
startup nomount pfile=database_path\INIT.ORA  
logfile group 1 'database_path\log1.ora' size 256K,  
logfile group 2 'database_path\log2.ora' size 256K  
datafile 'database_path\sys1.ora',  
'database_path\rbs1.ora',
```

```
'database_path\cim1.ora' ,  
'database_path\cim2.ora' ,  
'database_path\cim3.ora' ,  
'database_path\cim4.ora' ,  
'database_path\cim5.ora' ,  
'database_path\cim6.ora' ,  
'database_path\cim7.ora' ,  
'database_path\cim8.ora' ,  
'database_path\cim9.ora' ,  
'database_path\cim10.ora' ,  
'database_path\cim11.ora' ,  
'database_path\tmp1.ora'
```

Save the changes.

7 Load the restored database.

Setting Up the Inventory Database for MS SQL Server 2000

This section provides information on the following topics:

- ◆ [“Configuring the Inventory Database for MS SQL Server 2000” on page 701](#)
- ◆ [“Connecting the Inventory Server and ConsoleOne to the Inventory Database Running MS SQL 2000” on page 703](#)

Configuring the Inventory Database for MS SQL Server 2000

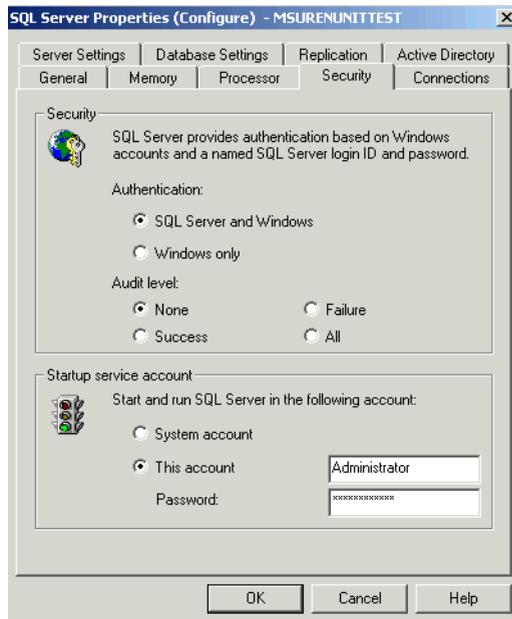
Prerequisites for configuring the database include the following:

- Microsoft SQL Server 2000 version 8.00.194 must be installed on the Windows NT/2000 server.
- Minimum free disk space of 50 MB to extract the p1mssqlinldb.zip file.
- Ensure that you have sufficient disk space to store the inventory data on the server that has the Inventory database.

To configure the Inventory database for MS SQL Server 2000:

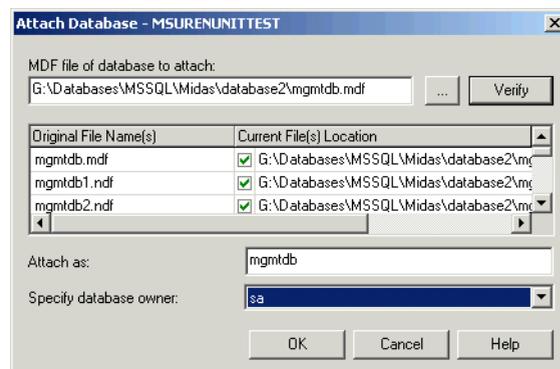
- 1** Copy the p1mssqlinldb.zip file from the *zenworks_for_servers_3_product_cd\zenworks\products\rminv\database\mssql* directory to *path_of_inventory_database_directory_on_the_database_server*.
- 2** Extract p1mssqlinldb.zip.
- 3** From the MS SQL server desktop Start menu, click Programs > Microsoft SQL Server > Enterprise Manager.
- 4** In the SQL Server Enterprise Manager, browse to Console Root/Microsoft SQL Servers/SQL Server Group/*machine_name_running_Inventory_database*.
- 5** Right -click *machine_name_running_Inventory_database*, then click Properties.

- 6** In the SQL Server Properties dialog box, click the Security tab and ensure that the authentication is set to SQL Server and Windows.



- 7** Click OK.
- 8** Browse to *machine_name_running_Inventory_database/Databases* and right-click Databases, click All Tasks, then double-click Attach Database.
- 9** In the Attach Database dialog box, do the following:
- 9a** Click the Browse button to browse to and select mgmtdb.mdf as the .mdf database file to be attached.
 - 9b** Ensure that the value of the Attach As field is mgmtdb.
 - 9c** Select sa from the Specify database owner drop-down list.
 - 9d** Click OK.

The ZENworks Inventory database (mgmtdb) is attached to the Databases server group.



- 10** Select mgmtdb, then click the Tools menu, click SQL Query Analyzer.

11 In the SQL Query Analyzer, do the following:

11a Ensure that mgmtdb is selected in the drop-down list.

11b Click File > Open.

11c Select the createloginnames.sql query file from
zenworks_for_servers_3_product_cd\zenworks\products\rminv\database\mssql
directory.

11d Click Query > Execute.

On successful execution, the following message is displayed in the Message pane:

New Login Created

12 Continue with “[Connecting the Inventory Server and ConsoleOne to the Inventory Database Running MS SQL 2000](#)” on page 703.

Connecting the Inventory Server and ConsoleOne to the Inventory Database Running MS SQL 2000

The Inventory server components and ConsoleOne use the Microsoft JDBC driver to connect to the Inventory database on MS SQL 2000. You must install and configure the Microsoft SQL Server 2000 driver for JDBC to use the Inventory system.

To configure the Microsoft SQL Server 2000 driver for JDBC to access the Inventory database running on MS SQL 2000:

1 Download the Windows English version of Microsoft JDBC driver from the [Microsoft SQL Server web site \(http://www.microsoft.com/sql/downloads/2000/jdbc.asp\)](http://www.microsoft.com/sql/downloads/2000/jdbc.asp).

2 Install the driver on a Windows machine.

3 Copy the msbase.jar, msutil.jar, and mssqlserver.jar files to the
inventory_server_installation_directory\inv\server\lib directory.

4 On all NetWare Inventory servers attached to the Inventory database mounted on MS SQL Server 2000, edit the *sys:\system\invenv.ncf* file to add the names of all the jar files of the JDBC driver in the following format:

```
envset tmpopath=$tmpopath;$root_dir\lib\msbase.jar
envset tmpopath=$tmpopath;$root_dir\lib\msutil.jar
envset tmpopath=$tmpopath;$root_dir\lib\mssqlserver.jar
...
...
envset tmpopath=$tmpopath;$root_dir\lib\jdbcdrv.zip
```

5 On all Windows NT/2000 Inventory servers attached to the Inventory database mounted on MS SQL Server 2000, do the following:

♦ Edit the *inventory_server_installation_directory\wminv\bin\zensetenv.ini* file to append the following entry at the end of each line containing the classpath:

```
..\..\lib\msbase.jar;..\..\lib\msutil.jar;..\..\lib\mssqlserver.jar;
```

♦ Edit the *inventory_server_installation_directory\wminv\bin\invenv.bat* file to add the following lines:

```
set tmpopath=%tmpopath%;..\..\lib\msbase.jar
```

```
set tmpopath=%tmpopath%;..\..\lib\msutil.jar
```

```
set tmpopath=%tmpopath%;..\..\lib\mssqlserver.jar
```

6 On the machine running ZfS ConsoleOne with Inventory snap-ins, copy the msbase.jar, msutil.jar, and mssqlserver.jar files to the *consoleone_installation_directory*\lib\zen directory.

7 In ConsoleOne, create a database object in the same container where the Inventory server is installed.

7a Right-click the container.

7b Click New, click Object, select ZENworks Database from the list of objects, then click OK.

7c Enter a name for the database object, then click OK.

8 Configure the Database server options of the Database object.

8a In ConsoleOne, right-click the database object, click Properties, then click the ZENworks Database tab.

8b Select the database server object using any of the following methods:

- ◆ If eDirectory is installed on the database server, in the Server DN field, browse for and select the Server object for the server where the database is physically installed and running.

The server's IP address is automatically populated to the Server IP Address or DNS Name drop-down list. If the selected server object has more than one IP address, select the appropriate IP address.

IMPORTANT: Ensure that the DNS name of the database server configured for the database object is valid. If the DNS name is invalid, you must select an appropriate database server IP address in the Database object property page.

- ◆ If eDirectory is not installed on the database server, specify the server's IP address or the DNS name in the Server IP Address or DNS Name field.

8c Type the values for the following options:

- ◆ **Database (Read-Write) User Name:** *MW_DBA*
- ◆ **Database (Read-Write) Password:** *novell*
- ◆ **Database (Read Only) User Name:** *MWM_READER*
- ◆ **Database (Read Only) Password:** *novell*
- ◆ **Database (Write Only) User Name:** *MWM_UPDATER*
- ◆ **Database (Write Only) Password:** *novell*

8d Click Apply.

8e To configure the JDBC Driver properties, click the JDBC Driver Information tab.

8f Select MS SQL, then click Default Settings.

This populates the fields with default JDBC driver information.

Modify the database settings based on the configuration of your MS SQL Server. The database settings for MS SQL are:

- ◆ **Driver:** *com.microsoft.jdbc.sqlserver.SQLServerDriver*
- ◆ **Protocol:** *jdbc:*

- ◆ **SubProtocol:** *microsoft*.
- ◆ **SubName:** *sqlserver://*
- ◆ **Port:** *1433*
- ◆ **Flags:** Not applicable for MS SQL
- ◆ **Database Service Name:** Not applicable for MS SQL

8g Click Apply, then click Close.

Configuring Inventory Servers for Server Inventory

Based on the role on the Inventory server, you need to configure the settings of the Inventory server.

You can set policies to control how Inventory servers collect inventory. The Inventory policy settings configure the inventory scanning options for the selected Distributed Server Inventory Package. The Inventory policy settings stored in eDirectory are associated with an inventoried server object. Each inventoried server object has an associated Inventory policy package.

The following table lists the policies that you should configure after installing Server Inventory:

To set up this type of Inventory server: Do this:	
Standalone Server	<ol style="list-style-type: none"> 1. Follow the steps in “Configuring the Server Inventory Policy” on page 707. 2. Follow the steps in “Configuring the Database Location Policy” on page 710.
Root Server	<ol style="list-style-type: none"> 1. Follow the steps in “Configuring the Inventory Service Object” on page 706. 2. Follow the steps in “Configuring the Database Location Policy” on page 710.
Root Server with Inventoried Servers	<ol style="list-style-type: none"> 1. Follow the steps in “Configuring the Inventory Service Object” on page 706. 2. Follow the steps in “Configuring the Server Inventory Policy” on page 707. 3. Follow the steps in “Configuring the Database Location Policy” on page 710.
Intermediate Server	<ol style="list-style-type: none"> 1. Follow the steps in “Configuring the Inventory Service Object” on page 706. 2. Follow the steps in “Configuring the Roll-Up Policy” on page 711.
Intermediate Server with Database	<ol style="list-style-type: none"> 1. Follow the steps in “Configuring the Inventory Service Object” on page 706. 2. Follow the steps in “Configuring the Roll-Up Policy” on page 711. 3. Follow the steps in “Configuring the Database Location Policy” on page 710.

To set up this type of Inventory server: Do this:

Intermediate Server with Database and Inventoried Servers	<ol style="list-style-type: none">1. Follow the steps in “Configuring the Inventory Service Object” on page 706.2. Follow the steps in “Configuring the Server Inventory Policy” on page 7073. Follow the steps in “Configuring the Roll-Up Policy” on page 7114. Follow the steps in “Configuring the Database Location Policy” on page 710.
Leaf Server with Database	<ol style="list-style-type: none">1. Follow the steps in “Configuring the Inventory Service Object” on page 706.2. Follow the steps in “Configuring the Server Inventory Policy” on page 707.3. Follow the steps in “Configuring the Roll-Up Policy” on page 711.4. Follow the steps in “Configuring the Database Location Policy” on page 710.
Leaf Server	<ol style="list-style-type: none">1. Follow the steps in “Configuring the Inventory Service Object” on page 706.2. Follow the steps in “Configuring the Server Inventory Policy” on page 707.3. Follow the steps in “Configuring the Roll-Up Policy” on page 711.
Intermediate Server with Inventoried Servers	<ol style="list-style-type: none">1. Follow the steps in “Configuring the Inventory Service Object” on page 706.2. Follow the steps in “Configuring the Server Inventory Policy” on page 707.3. Follow the steps in “Configuring the Roll-Up Policy” on page 711.

IMPORTANT: After installing and configuring Server Inventory, you must run the Inventory Services on the Inventory server. For more information, see “[Starting the Inventory Service](#)” on page 712.

Configuring the Inventory Service Object

The Inventory Service object settings configure the scanning for the associated inventoried servers. From the Inventory Service Object property page, you can configure the following:

- ◆ [Inventory Server Role](#)
- ◆ [Discard Scan Data Time](#)
- ◆ [Scan Directory Path](#)
- ◆ [Enable Scan](#)

To open the Inventory Service Object properties page:

- 1 In ConsoleOne, right-click the Inventory Service object (*servername_ZenInvservice*), click Properties, then click the Inventory Service Object Properties tab.
- 2 Modify the following settings:

Inventory Server Role: Based on the Inventory servers that you have deployed for scanning inventory, you must specify the role of the Inventory server. See [“Understanding the Inventory Server Roles” on page 661](#).

Discard Scan Data Time: Any scan data files (.zip files) that have scan information collected before the Discard Scan Data Time that you specify in the Inventory Service Object Property page will be discarded. The scan data files are removed from the Inventory server, which is one of the following types: Intermediate Server, Intermediate Server with Database, Intermediate Server with Database and Inventoried Servers, and Intermediate Server with Inventoried Servers.

Scan Directory Path: Specify the volume or the directory of the Scan Directory (*scandir*) setting in the Inventory Service Object property page. The *scandir* directory path is the location on the Inventory server that stores the scan data files (.str files). The format of the Scan Directory Path is as follows: *Inventory_server_name\volume_of_the_server_directory*.

You cannot modify the Inventory Server name specified in the *scandir* path. But if you want modify the directory, make sure that the new directory already exists before changing the *scandir* path.

To modify the path: click the Browse button.

- ◆ On NetWare, click Browse to select and add the path or enter the path.
- ◆ On Windows, you must manually enter the path.

Enable Scan: To scan the inventoried servers associated with the Inventory Service object, you must enable the scan option listed in the Inventory Service Object property page. To disable the scanning of inventoried servers, deselect this option.

- 3 Click OK.

NOTE: If you are modifying the Inventory policies or configuring the objects, always stop the Inventory services. Configure the policies and properties of the objects. Restart the Inventory services again. For more information, see [“Starting and Stopping the Inventory Service” on page 712](#).

Configuring the Server Inventory Policy

The Server Inventory policy contains the IP address or the DNS name of the Inventory server to which the inventory data will be sent. This policy also contains the inventory scanning schedule for the associated inventoried server. You must configure the Server Inventory policy for each inventoried server.

To configure the Server Inventory policy:

- 1 In ConsoleOne, right-click the Distributed Server Package, click Properties, then click Policies.
- 2 Click Policies > General, NetWare, or Windows sub-options.

To configure for both NetWare and Windows inventoried servers, click the General sub-option.

NOTE: Do not select to configure policies in the Solaris or the Linux suboption as they are not supported.

- 3** Select the check box under the Enabled column for the Server Inventory policy.
- 4** Click Properties > the Server Inventory Policy tab.
- 5** Browse to select the DN of the Inventory Service object.

This setting specifies that the scanner will send the server scan data to this Inventory server.

NOTE: The Inventory Service object must be in the same eDirectory tree as the Server Inventory policy.

- 6** Select the DNS name or the IP address of the Inventory server.
- 7** If you want to send or roll-up the scan data to an Inventory server that is across the firewall, specify the IP address and the port number of the proxy server.
- 8** (Optional) To customize Inventory scanning, do the following:

- 8a** Click the Hardware Scan tab.

This tab will be displayed while configuring the Server Inventory policy of the Windows servers.

- 8b** Select the Enable DMI Scan option to include DMI scanning of inventoried Windows servers.
- 8c** Select the Enable WMI Scan option to include WMI scanning of inventoried Windows servers.
- 8d** Click the Software Scan tab.
- 8e** Select the Enable the Software Scan option to enable software scanning of inventoried servers.
- 8f** Click the Custom Scan Editor button to select the software that you want to scan for at the servers, then modify the list.

The Inventory scanner scans for the application information configured in the Custom Scan Editor.

For example, specify the following information in the Custom Scan Editor: Vendor Name=Microsoft; Product Name=Microsoft Office - Word; Product Version=10.0; FileName=winword.exe; File Size=10000.

In the Custom Scan Editor dialog box, the Inventory scanner scans for the winword.exe file having a size of 10000 bytes on the inventoried servers associated with the Server Inventory policy. If the file is found, the scanner stores "Microsoft;Microsoft Office - Word;10.0" for "winword.exe;10000" in the Inventory database.

The Custom Scan table lists the applications information that you want to scan for at the inventoried server such as vendor name, product name, product version, filename, and file size (in bytes).

You can enter a maximum of 1000 entries to the Custom Scan table.

Use the following methods to add the information of the applications to the Custom Scan table:

- ◆ Adding new entries
 1. Click Add.
 2. Specify the details.

You can specify details for one of the following attributes: Vendor Name, Product Name, or File\Location Name. But if you specify Product Version or File Size, you must also

configure at least one of the following: Vendor Name, Product Name, or File/Location Name.

3. Click OK.

◆ Importing the contents from an existing file

1. Open a text editor and create a file with the following contents:

number_of_entries;number_of_columns

vendor_name;product_name;product_version;file_name;file_size_in_bytes;

...

vendor_name;product_name;product_version;file_name;file_size_in_bytes;

Fill in all the details of the application. The separator between the columns is a semicolon. Save the file as a text file with any extension you prefer. You can then import the file into the Custom Scan table.

A sample file is as follows:

2;5

Microsoft;Microsoft Office-Word;10.0;winword.exe;10000

Microsoft;Microsoft Office-Excel;5.0;excel.exe;50000

2. In the Custom Scan Editor dialog box, click Import.

3. Browse and select the file.

4. Click Open.

8g Select the Product Identification Number option to include scanning of product identification numbers of the Microsoft applications installed on the inventoried Windows servers.

8h Click the Configuration Editor tab.

8i Select the appropriate sub-option; Asset Information, Zipped Names, or SWRules.

8j Click Set Default to get the default settings.

8k If required, modify the settings of the configuration files, then click OK.

9 Click the Policy Schedule tab.

10 Modify the schedule, click Apply, then click Close.

11 From the Distributed Server Package property page, click the Distribution tab, then click Add.

12 Browse to add the Distribution object, then click OK.

13 Click Apply, then click close.

14 In ConsoleOne, right-click the Inventory Service object (*servername_ZenInvService*), click Properties, then click the Inventory Service Object Properties tab.

15 Make sure the Enable Scan of Machines check box is selected, then click OK.

This setting ensures that scanning is enabled for the servers associated with the selected Inventory server.

Configuring the Database Location Policy

The Database Location policy contains the location of the Inventory database. You can associate the Database object with a container under which the Inventory Service object is located through using the Service Location Package or with an Inventory server through using the Server Package.

NOTE: If you configure the Service Location Package and the Server Package, the Server Package settings will override the Service Location Package settings.

To associate the Database object with a container under which the Inventory Service object is located:

- 1** In ConsoleOne, right-click the Service Location Package, click Properties, then click Policies.
- 2** Select the check box under the Enabled column for the ZENworks Database policy.
- 3** Click Properties.
- 4** Click the Inventory Management tab.
- 5** Browse to the DN of the Inventory Database object, then click OK.

For a Sybase database, the database object is automatically created during the Server Inventory installation unless you are installing on a Windows NT/2000 server without eDirectory installed. To manually create the database object, see [“Manually Creating the Inventory Database Object for Sybase” on page 685](#).

For an Oracle database, you must create the database object and configure the object. For more information, see [“Setting Up the Inventory Database for Oracle8i and Oracle9i” on page 690](#).

- 6** Click OK.
- 7** Click the Associations tab, then click Add.
- 8** Browse to select the container under which the Inventory Service object is located, then click OK.
- 9** Click Apply, then click Close.

To associate the Database object with an Inventory server:

- 1** In ConsoleOne, right-click the Server Package, click Properties, then click Policies.
- 2** Select the check box under the Enabled column for the ZENworks Database policy.
- 3** Click Properties.
- 4** Click the Inventory Management tab.
- 5** Browse to the DN of the Inventory Database object, then click OK.

For a Sybase database, the database object is automatically created during the Server Inventory installation unless you are installing on a Windows NT/2000 server without eDirectory installed. To manually create the database object, see [“Manually Creating the Inventory Database Object for Sybase” on page 685](#).

For an Oracle database, you must create the database object and configure the object. For more information, see [“Setting Up the Inventory Database for Oracle8i and Oracle9i” on page 690](#).

- 6** Click OK.
- 7** Click the Associations tab, then click Add.

- 8 Browse to select an Inventory server object, then click OK.
- 9 Click Apply, then click Close.

NOTE: If you are modifying the Inventory policies or configuring the objects, always stop the Inventory services. Configure the policies and properties of the objects. Restart the Inventory services again. For more information, see [“Starting and Stopping the Inventory Service” on page 712](#).

Configuring the Roll-Up Policy

The Roll-Up policy settings configure the selected Inventory server for roll-up of scan information. The settings in the Roll-Up policy identify the next-level Inventory server (DN of the Inventory Service object) for moving the scan data from the selected Inventory server. These settings stored in eDirectory are associated with the Inventory Server object.

To configure the Roll-Up policy:

- 1 In ConsoleOne, right-click the Policy Packages container, click New > Policy Package > Server Package > RollupPolicy > Next.
- 2 Type a name for the Server Package, click Next, then click Finish.
- 3 In ConsoleOne, right-click the Server Package, click Properties, click Policies, then click General.
- 4 Check the check box under the Enabled column for the Rollup Policy.
- 5 Click Properties.
- 6 Click the Roll-up Policy tab, then click Roll-up Policy.
- 7 Browse to select the DN of the Inventory Service object, then click OK.

Destination Server Object: You must specify the DN of the Inventory Service object at the next level Inventory server for moving the scan data from the selected Inventory server. The server that you specify must be another Intermediate Server, Intermediate Server with Database, Intermediate Server with Database and Inventoried Servers, Intermediate Server with Inventoried Servers, Root Server, or Root Server with Inventoried Servers.

NOTE: Ensure that the specified Inventory server is a different server because you cannot roll-up of data to the same Inventory server. Also, you cannot specify the lower-level Inventory server as the next-destination server for roll-up of data.

- 8 Select the IP address or the DNS name of the next level Inventory server.
- 9 If the roll-up is to an Inventory server that is across the firewall, specify the IP address or the DNS name and the port number of the proxy server.
- 10 Click the Associations tab, then click Add.

The first time you enable the Roll-Up policy, you will be prompted to associate the policy package. The policy you configured and enabled earlier will not be in effect until you associate this policy package with an Inventory server. Browse for the Inventory server that you want to associate the Roll-Up policy to, click OK > OK.

- 11 In ConsoleOne, right-click the Server Package, click Properties, then click Policies. Click NetWare or click Window sub-options.
- 12 Click the Roll-Up Policy row > Properties > Roll-Up Policy tab > Roll-Up Schedule. Modify the settings for scheduling the roll-up time, then click OK.

When you schedule the roll-up of data in the Inventory policies, we recommend the roll-up frequency should be at least one day. If the roll-up of scan data is scheduled too frequently,

for example less than one hour, there may be some performance degradation of the Inventory server.

NOTE: If you are modifying the Inventory policies or configuring the objects except for the Roll-Up schedule, always stop the Inventory services. Configure the policies and properties of the objects. Restart the Inventory services again. For more information, see [“Starting and Stopping the Inventory Service” on page 712](#).

Starting and Stopping the Inventory Service

The section provides information on:

- ♦ [“Starting the Inventory Service” on page 712](#)
- ♦ [“Stopping the Inventory Service” on page 713](#)

Starting the Inventory Service

Before you start the Inventory service, make sure that the TED components and the Inventory database are up and running. The Inventory database will be automatically started after the installation.

To manually start the Inventory services on the NetWare Inventory server, enter **startinv** at the server console prompt.

To manually start the Inventory services on the Windows NT Inventory server:

- 1** In the Control Panel, double-click Services.
- 2** Select Novell ZEN Inventory, then click Start.

To manually start the Inventory services on the Windows 2000 Inventory server:

- 1** In the Control Panel, double-click Administrative Tools.
- 2** Double-click Services.
- 3** Select Novell ZEN Inventory, then click Start.

To start a service on Windows NT/2000 server from the console prompt:

- 1** Go to the *Installation_directory*\inv\server\wminv\bin directory.
- 2** At the prompt, enter **StartSer service_name**.
where *service_name* refers to an Inventory service.

After starting the Inventory service, make sure that the Inventory services are up and running.

To list all services:

- ♦ On a NetWare Inventory server, enter **ListSer *** at the console prompt.
- ♦ On a Windows NT/2000 Inventory server, enter **ListSer "*"** at the console prompt.

If the services are not up and running, check the Server Status log. For more information on the Server Status log, see [“Viewing the Status of Inventory Components on an Inventory Server” on page 844](#).

Stopping the Inventory Service

To stop the Inventory services on the NetWare Inventory server:

- ♦ To stop an Inventory service, enter **stopser *Inventory_service_name*** at the server console prompt.
- ♦ To stop all the Inventory services, enter **stopser * at the server console prompt.**

To stop the Inventory services on the Windows NT Inventory server:

- 1** In the Control Panel, double-click Services.
- 2** Select Novell ZEN Inventory, then click Stop.

To stop the Inventory services on the Windows 2000 Inventory server:

- 1** In the Control Panel, double-click Administrative Tools.
- 2** Double-click Services.
- 3** Select Novell ZEN Inventory, then click Stop.

To stop a service on Windows NT/2000 servers from the console prompt:

- 1** Go to the *Installation_directory\inv\server\wminv\bin* directory.
- 2** Enter **StopSer *service_name***.

where *service_name* refers to an Inventory service.

Changing the Role of the Inventory Server

When you install ZfS, by default, the role of the Inventory server is a Standalone Server. By configuring the Inventory Service object, you can assign specific roles to the Inventory server based on your inventory deployment.

For example, if the deployment plan identifies three Inventory servers, such as a Root Server, an Intermediate Server with Database, and a Leaf Server for inventory deployment, you install Server Inventory on these servers, and choose the role for the Inventory server. Later, if you want to make changes in the inventory deployment, such as attaching the inventoried servers to the existing Root Server, you need to change the role of the Inventory Service object from Root Server to Root Server with Inventoried Servers. Additionally, depending on the new role, there are some policies you need to configure.

To change the role for any Inventory server:

- 1** Plan the change of roles carefully because the changes will impact the existing inventory deployment. Also, consider the disk space requirements and ensure that you have the required configurations for Inventory.
- 2** In ConsoleOne, right-click the Inventory Service object (*servername_ZenInvservice*), click Properties, then click the Inventory Service Object Properties tab.
- 3** Choose the new role of the Inventory Service object, then click Apply.

You will see a list of actions that you should follow based on the chosen role. For example, if you change the Root Server to a Root Server with Inventoried Servers, you need to configure the Server Inventory policy for the inventoried servers that you have attached. Similarly, to change the role to any other Inventory server, you need to follow the instructions to make the

new role change effective. For more information, see [“Configuring Inventory Servers for Server Inventory” on page 705](#).

- 4 Bring down the services running on the changed Inventory server, follow the actions that you need to change the role, and then bring up the Inventory services.

To stop all Inventory Services:

- ◆ At NetWare server console prompt, enter the following commands:

```
stopser *  
java -killZenWSInv
```

- ◆ On the Windows NT/2000 server, from the Services window, click Novell ZEN Inventory > Stop.

To restart all Inventory Services:

- ◆ At NetWare server console prompt, enter **startinv**
- ◆ On the Windows NT/2000 server, from the Services window, click Novell ZEN Inventory > Start.

The following sections contain information to help you change the role of the Inventory Service object:

- ◆ [“Changing the Role of the Root Server” on page 715](#)
- ◆ [“Changing the Role of the Root Server with Inventoried Servers” on page 716](#)
- ◆ [“Changing the Role of the Intermediate Server” on page 717](#)
- ◆ [“Changing the Role of the Intermediate Server with Database” on page 718](#)
- ◆ [“Changing the Role of the Intermediate Server with Database and Inventoried Servers” on page 719](#)
- ◆ [“Changing the Role of the Intermediate Server with Inventoried Servers” on page 720](#)
- ◆ [“Changing the Role of the Leaf Server” on page 721](#)
- ◆ [“Changing the Role of the Leaf Server with Database” on page 722](#)
- ◆ [“Changing the Role of the Standalone Server” on page 722](#)

Changing the Role of the Root Server

To change the role of the Root Server to a different role, follow the actions specified in the following table:

To change the role of the Root Server to ...	Tasks:
Root Server with Inventoried Servers	Perform the following task: <ol style="list-style-type: none">1. After changing the role, configure the Server Inventory policy so that the inventoried servers that you have attached to the Root Server with Inventoried servers will be scanned for.
Intermediate Server	Perform the following tasks: <ol style="list-style-type: none">1. Before changing the role, remove the Database Location policy associated with a Root Server.2. After changing the role, configure the Roll-Up policy to specify the next-destination server for roll-up of data from this Inventory server.
Intermediate Server with Database	Perform the following task: <ol style="list-style-type: none">1. After changing the role, configure the Roll-Up policy to specify the next-destination Inventory server for roll-up of data from this Inventory server.
Intermediate Server with Database and Inventoried Servers	Perform the following tasks after changing the role: <ol style="list-style-type: none">1. Configure the Server Inventory policy so that the inventoried servers that you have attached will be scanned for.2. Configure the Roll-Up policy to specify the next-destination server for roll-up of data from this Inventory server.
Intermediate Server with Inventoried Servers	Perform the following tasks: <ol style="list-style-type: none">1. Before changing the role, remove the Database Location policy associated with the Root Server.2. After changing the role, configure the Server Inventory policy so that the inventoried servers that you have attached will be scanned for.3. After changing the role, configure the Roll-Up policy to specify the next-destination Inventory server for roll-up of data from this Inventory server.
Leaf Server, Leaf Server with Database, or Standalone Server	Server Inventory does not allow you to change the Root Server to these Inventory servers because these changes affect the complete inventory system. If you want to assign these roles, you should reinstall and set up the Server Inventory component.

Changing the Role of the Root Server with Inventoried Servers

Follow the actions specified in the following table:

To change the role of the Root Server with Inventoried Servers to ...	Tasks:
Root Server	Perform the following task: <ol style="list-style-type: none">1. Before changing the role, remove the Server Inventory policy associated with the Root Server with Inventoried Servers.
Intermediate Server	Perform the following tasks: <ol style="list-style-type: none">1. Before changing this role, remove the Database Location policy and the Server Inventory policy.2. After changing the role, configure the Roll-Up policy to specify the next-destination server for roll-up of data from this Inventory server.
Intermediate Server with Database	Perform the following tasks: <ol style="list-style-type: none">1. Before changing the role, if the Server Inventory policy is associated with the Root Server with Inventory servers, remove the policy for those servers attached to this Inventory server or to the lower-level Inventory servers that roll up to this Inventory server.2. After changing the role, configure the Roll-Up policy to specify the next-destination server for roll-up of data from this Inventory server.
Intermediate Server with Database and Inventoried Servers	Perform the following task: <ol style="list-style-type: none">1. After changing the role, configure the Roll-Up policy to specify the next-destination Inventory server for roll-up of data from this Inventory server.
Intermediate Server with Inventoried Servers	Perform the following task: <ol style="list-style-type: none">1. Before changing the role, remove the Database Location policy that is associated with the Root Server with Inventoried Servers.
Leaf Server, Leaf Server with Database, or Standalone Server	Server Inventory does not allow you to change the Root Server to these Inventory servers because these changes affect the complete inventory system. If you want to assign these roles, you should reinstall and set up the Server Inventory component.

Changing the Role of the Intermediate Server

Follow the actions specified in the following table:

To change the role of the Intermediate Server to ...	Tasks:
Root Server	Perform the following tasks: <ol style="list-style-type: none"><li data-bbox="786 350 1376 385">1. Before changing the role, remove the Roll-Up policy.<li data-bbox="786 397 1422 451">2. After changing the role, configure the Database Location policy.
Root Server with Inventory Servers	Perform the following tasks: <ol style="list-style-type: none"><li data-bbox="786 532 1376 566">1. Before changing the role, remove the Roll-Up policy.<li data-bbox="786 578 1453 661">2. After changing the role, configure the Server Inventory policy for those inventoried servers attached to this server and the Database Location policy.
Intermediate Server with Database	Perform the following task: <ol style="list-style-type: none"><li data-bbox="786 733 1422 792">1. After changing the role, configure the Database Location policy for this Inventory server.
Intermediate Server with Database and Inventoried Servers	Perform the following tasks: <ol style="list-style-type: none"><li data-bbox="786 864 1453 1010">1. After changing the role, configure the Server Inventory policy so that all the inventoried servers associated to this Inventory Service object, and also those inventoried servers associated to the lower-level Inventory servers that roll up to this Inventory server will be scanned for.<li data-bbox="786 1022 1422 1084">2. After changing the role, configure the Database Location policy.
Intermediate Server with Inventoried Servers	Perform the following task: <ol style="list-style-type: none"><li data-bbox="786 1157 1453 1243">1. After changing the role, configure the Server Inventory policy so that the inventoried servers that you have attached will be scanned for.
Leaf Server, Leaf Server with Database, or Standalone Server	Server Inventory does not allow you to change the Intermediate Server to these Inventory servers because these changes affect the complete inventory system. If you want to assign these roles, you should reinstall and set up the Server Inventory component.

Changing the Role of the Intermediate Server with Database

Follow the actions specified in the following table:

To change the role of the Intermediate Server with Database to ... Tasks:	
Root Server	Perform the following task: <ol style="list-style-type: none">1. Before changing the role, remove the Roll-Up policy associated with the Intermediate Server with Database.
Root Server with Inventoried Servers	Perform the following tasks: <ol style="list-style-type: none">1. Before changing the role, remove the Roll-Up policy associated with the Intermediate Server with Database.2. After changing the role, configure the Server Inventory policy so that the inventoried servers that you have attached will be scanned for.
Intermediate Server	Perform the following task: <ol style="list-style-type: none">1. Before changing the role, remove the Database Location policy that is associated with the Intermediate Server with Database.
Intermediate Server with Database and Inventoried Servers	Perform the following task: <ol style="list-style-type: none">1. After changing the role, configure the Server Inventory policy so that the inventoried servers attached will be scanned for.
Intermediate Server with Inventoried Servers	Perform the following tasks: <ol style="list-style-type: none">1. Before changing the role, remove the Database Location policy that is associated with the Intermediate Server with Database.2. After changing the role, configure the Server Inventory policy so that the inventoried servers that you have attached will be scanned for.
Leaf Server, Leaf Server with Database, or Standalone Server	Server Inventory does not allow you to change the Intermediate Server to these Inventory servers because these changes affect the complete inventory system. If you want to assign these roles, you should reinstall and set up the Server Inventory component.

Changing the Role of the Intermediate Server with Database and Inventoried Servers

Follow the actions specified in the following table:

To change the role of the Intermediate Server with Database and Inventoried Servers to ...	Tasks:
Root Server	Perform the following tasks before changing the role: <ol style="list-style-type: none"> 1. Remove the Roll-Up policy associated with the Intermediate Server with Database and Inventoried Servers. 2. Remove the Server Inventory policy associated with the inventoried server so that the inventoried servers will not send the scan files to this server.
Root Server with Inventoried Servers	Perform the following task: <ol style="list-style-type: none"> 1. Before changing the role, remove the Roll-Up policy associated with the Intermediate Server with Database and Inventoried Servers.
Intermediate Server	Perform the following tasks before changing the role: <ol style="list-style-type: none"> 1. Remove the Server Inventory policy associated with the lower-level servers that roll up to the Intermediate Server with Database and Inventoried Servers. 2. Remove the Database Location policy associated with the Intermediate Server with Database and Inventoried Servers.
Intermediate Server with Database	Perform the following task: <ol style="list-style-type: none"> 1. Remove the Server Inventory policy of the Intermediate Server with Database and Inventoried Servers or reconfigure the policy.
Intermediate Server with Inventoried Servers	Perform the following task: <ol style="list-style-type: none"> 1. Before changing the role, remove the Database Location policy associated with the Intermediate Server with Database and Inventoried Servers.
Leaf Server, Leaf Server with Database, Standalone Server	Server Inventory does not allow you to change the Intermediate Server to these servers because these changes affect the complete inventory system. If you want to assign these roles, you should reinstall and set up the Server Inventory component.

Changing the Role of the Intermediate Server with Inventoried Servers

Follow the actions specified in the following table:

To change the role of the Intermediate Server with Inventoried Servers to ... Tasks:	
Root Server	<p>Perform the following tasks:</p> <ol style="list-style-type: none">1. Before changing the role, remove the Roll-Up policy associated with the Intermediate Server with Inventoried Servers.2. Before changing the role, remove the Server Inventory policy associated with the inventoried server so that the inventoried servers attached will not send the scan files to this Inventory server.3. After changing the role, configure the Database Location policy for this Inventory server.
Root Server with Inventoried Servers	<p>Perform the following tasks:</p> <ol style="list-style-type: none">1. Before changing the role, remove the Roll-Up policy associated with the Intermediate Server with Inventoried Servers.2. After changing the role, configure the Server Inventory policy for those inventoried servers attached to the lower-level Inventory server that roll up to this Inventory server.3. After changing the role, configure the Database Location policy.
Intermediate Server	<p>Perform the following task:</p> <ol style="list-style-type: none">1. Before changing the role, remove the Server Inventory policy.
Intermediate Server with Database	<p>Perform the following tasks:</p> <ol style="list-style-type: none">1. Before changing the role, remove the Server Inventory policy associated to the inventoried server attached to this Inventory Service object.2. After changing the role, configure the Database Location policy for this Inventory server.
Intermediate Server with Database and Inventoried Servers	<p>Perform the following task:</p> <ol style="list-style-type: none">1. After changing the role, configure the Database Location policy for this Inventory server.
Leaf Server, Leaf Server with Database or Standalone Server	<p>Server Inventory does not allow you to change the Intermediate Server to these Inventory servers because these changes affect the complete inventory system. If you want to assign these roles, you should reinstall and set up the Server Inventory component.</p>

Changing the Role of the Leaf Server

Follow the actions specified in the following table:

To change the role of the Leaf Server to ...	Tasks:
Root Server	Perform the following tasks: <ol style="list-style-type: none">1. Before changing the role, remove the Roll-Up policy associated with the Leaf Servers.2. Before changing the role, remove the Server Inventory policy associated with the inventoried server.3. After changing the role, configure the Database Location policy for the Root Server.
Root Server with Inventoried Servers	Perform the following tasks: <ol style="list-style-type: none">1. Before changing the role, remove the Roll-Up policy associated with the Leaf Server.2. After changing the role, configure the Database Location policy for the Root Server with Inventoried Servers.
Intermediate Server	Perform the following tasks: <ol style="list-style-type: none">1. Before changing the role, remove the Server Inventory policy for those inventoried servers associated with the Inventory server or reconfigure the policy.
Intermediate Server with Database	Perform the following tasks: <ol style="list-style-type: none">1. Before changing the role, remove the Server Inventory policy for those inventoried servers associated with the lower-level Inventory servers that roll up to this Inventory server or reconfigure the policy.2. After changing the role, configure the Database Location policy for this Inventory server.
Intermediate Server with Database and Inventoried Servers	Perform the following task: <ol style="list-style-type: none">1. After changing the role, configure the Database Location policy for this Inventory server.
Intermediate Server with Inventoried Servers	This change of role does not require any specific policy modifications.
Leaf Server with Database	Perform the following task: <ol style="list-style-type: none">1. After changing the role, configure the Database Location policy for this Inventory server.
Standalone Server	Perform the following task: <ol style="list-style-type: none">1. Before changing the role, remove the Roll-Up policy associated with the Leaf Server.

Changing the Role of the Leaf Server with Database

Follow the actions specified in the following table:

To change the role of the Leaf Server with Database to ...	Tasks:
Root Server	Perform the following tasks before changing the role: <ol style="list-style-type: none">1. Remove the Server Inventory policy associated with the Leaf Server with Database.2. Remove the Roll-Up policy associated with the Leaf Server with Database.
Root Server with Inventoried Servers	Perform the following task: <ol style="list-style-type: none">1. Before changing the role, remove the Roll-Up policy associated with the Leaf Server with Database.
Intermediate Server	Perform the following task: <ol style="list-style-type: none">1. Before changing the role, remove the Server Inventory policy and the Database Location policy associated with the Leaf Server with Database.
Intermediate Server with Database	Perform the following task: <ol style="list-style-type: none">1. Before changing the role, remove the Server Inventory policy associated with the Leaf Server with Database.
Intermediate Server with Database and Inventoried Servers	This change of role does not require any specific policy modifications.
Intermediate Server with Inventoried Servers	Perform the following task: <ol style="list-style-type: none">1. Before changing the role, remove the Database Location policy associated with the Leaf Server with Database.
Leaf Server	Perform the following task: <ol style="list-style-type: none">1. Before changing the role, remove the Database Location policy associated with the Leaf Server with Database.
Standalone Server	Perform the following task: <ol style="list-style-type: none">1. Before changing the role, remove the Roll-Up policy associated with the Leaf Server with Database.

Changing the Role of the Standalone Server

Follow the actions specified in the following table:

To change the role of the Standalone Server to ...	Tasks:
Root Server	Perform the following task: <ol style="list-style-type: none">1. Before changing the role, remove the Server Inventory policy associated with the Standalone Server.
Root Server with Inventoried Servers	This change of role does not require any specific policy modifications.

To change the role of the Standalone Server to ...	Tasks:
Intermediate Server	Perform the following tasks: <ol style="list-style-type: none"> 1. Before changing the role, remove the Server Inventory policy and the Database Location policy associated with the Standalone Server. 2. After changing the role, configure the Roll-Up policy to specify the next-destination Inventory server for roll-up of data from the Intermediate Server with Database.
Intermediate Server with Database	Perform the following tasks: <ol style="list-style-type: none"> 1. Before changing the role, remove the Server Inventory policy associated with the Standalone Server. 2. After changing the role, configure the Roll-Up policy to specify the next-destination Inventory server for roll-up of data from the Intermediate Server with Database.
Intermediate Server with Database and Inventoried Servers	Perform the following tasks: <ol style="list-style-type: none"> 1. After changing the role, configure the Roll-Up policy to specify the next-destination server for roll-up of data from the Intermediate Server with Database and Inventoried Servers.
Intermediate Server with Inventoried Servers	Perform the following tasks: <ol style="list-style-type: none"> 1. Before changing the role, remove the Database Location policy associated with the Standalone Server. 2. After changing the role, configure the Roll-Up policy to specify the next-destination server for roll-up of data from the Intermediate Server with Inventoried Servers.
Leaf Server	Perform the following tasks: <ol style="list-style-type: none"> 1. Before changing the role, remove the Database Location policy associated with the Standalone Server. 2. After changing the role, configure the Roll-Up policy to specify the next-destination server for roll-up of data from the Leaf Server.
Leaf Server with Database	Perform the following task: <ol style="list-style-type: none"> 1. After changing the role, configure the Roll-Up policy to specify the next-destination server for roll-up of data from the Leaf Server with Database.

27

Understanding the Server Inventory Components

The following sections describe the Novell® ZENworks® for Servers (ZfS) Server Inventory components and processes:

- ◆ “Understanding the Inventory Service Manager” on page 725
- ◆ “Understanding the Server Configuration Service” on page 730
- ◆ “Understanding the Inventory Scanner” on page 730
- ◆ “Understanding the Sender-Receiver” on page 751
- ◆ “Understanding the Selector” on page 756
- ◆ “Understanding the Storer” on page 757
- ◆ “Understanding the Inventory Removal Service” on page 757
- ◆ “Understanding the Upgrade Service” on page 759
- ◆ “An Overview of the Inventory Components on the Inventory Server” on page 760
- ◆ “Understanding the Inventory Database” on page 760

Understanding the Inventory Service Manager

The Inventory Service Manager loads the inventory components on the Inventory server, based on the configuration parameters specified in the Inventory server properties file.

This section contains the following:

- ◆ “List of Services” on page 725
- ◆ “Services on NetWare Inventory Servers” on page 728
- ◆ “Services on Windows NT/2000 Inventory Servers” on page 729

List of Services

The Service Manager loads the following services:

Service Name	Description
Server Configuration Service	Loads the server configuration services
Inventory Scheduler Service	Loads the Inventory Scheduler
Selector Service	Loads the Selector

Service Name	Description
Receiver Service	Loads the Receiver
Sender Service	Loads the Sender
Storer Service	Loads the Storer
Scan Collector	Stores the .str files to the SCANDIR

Property File: There are property files that load the different services on the Inventory server depending on the role of the Inventory server. The name of the property file indicates the role of the Inventory server. Only the required services are loaded as per the role of the Inventory server. The property files should not be modified.

A sample role-based property file for a Leaf Server with Database is as follows:

```
[Server Configuration Service]
type = system
Load Sequence = 0
Load Option = auto
Class Name = com.novell.zenworks.desktop.inventory.
             servercommon.ServerConfig
Arguments =

[Upgrade Service]type = userLoad Sequence = 1Load Option =
                autoClass Name = com.novell.zenworks.desktop.inventory.
                upgradeService.UpgradeServiceArguments =

[Inventory Scheduler Service]
type = system
Load Sequence = 2
Load Option = auto
Class Name = com.novell.zenworks.desktop.inventory.
             servercommon.InventorScheduler
Arguments =

[Selector Service]
type = user
Load Sequence = 3
Load Option = auto
Class Name = com.novell.zenworks.desktop.inventory.
             selector.SelectorServiceInit
Arguments =

[Storer Service]
type = user
Load Sequence = 4
```

```
Load Option = auto
Class Name = com.novell.zenworks.desktop.inventory.
    storer.StorerServiceInit
Arguments =
[Sender Service]
type = user
Load Sequence = 5
Load Option = auto
Class Name = com.novell.zenworks.desktop.inventory.
    senderreceiver.control.SenderServiceInit
Arguments =
[NDSLookupForDB Service]
type = user
Load Sequence = 6
Load Option = manual
Class Name = com.novell.zenworks.desktop.inventory.
    dbutilities.NDSLookupForDB
Arguments = "WSDELETE.LOK"
[DBDelete Service]
type = user
Load Sequence = 7
Load Option = manual
Class Name = com.novell.zenworks.desktop.inventory.
    dbutilities.DBDelete
Arguments = "WSDELETE.LOK"
[DBBackup Service]
type = user
Load Sequence = 8
Load Option = manual
Class Name = com.novell.zenworks.desktop.inventory.
    dbutilities.DBBackup
Arguments = "Backup"
```

Do not modify these property files as services or the Service Manager cannot be loaded.

Depending on the role of the Inventory server, the server properties files include:

Server Type	Server Property File
Root Server	ROOT_DB.PROPERTIES
Root Server with Inventoried Servers	ROOT_DB_WKS.PROPERTIES
Intermediate Server	INT.PROPERTIES
Intermediate Server with Inventoried Servers	INT_WKS.PROPERTIES
Intermediate Server with Database	INT_DB.PROPERTIES
Intermediate Server with Database and Inventoried Servers	INT_DB_WKS.PROPERTIES
Leaf Server	LEAF_WKS.PROPERTIES
Leaf Server with Database	LEAF_DB_WKS.PROPERTIES
Standalone Server	STANDALONE.PROPERTIES

The Inventory Service Manager reads the server properties file (`config.properties`) and the role-based property file in the `Inventory_server_installation_directory\public\zenworks\wminv\properties` directory, and loads the required services and server components.

The contents of the `config.properties` file are as follows:

```
NDSTREE=treename
INVENTORYSERVICEDN=dn_of_the_inventory_service_object
SINGLETONPORT=65433
StoreRolledupAuditData=false
LDAPServerIP=LDAPserver_IPaddress
LDAPPort=LDAPserver_Portnumber
```

Services on NetWare Inventory Servers

On a NetWare[®] Inventory server, the installation program modifies the `autoexec.ncf` file located in `sys:\system` directory to load `startinv.ncf`. The `startinv.ncf` file located in the `sys:\system` brings up the Inventory Service Manager at Inventory server startup time.

The contents of the `startinv.ncf` file are as follows:

```
search add sys:\java\njclv2\bin
InvEnv
java -envDISPLAY=127.0.0.1:0 -sn"ZENworks Inventory Service"
  -noclassgc -DConfigFile=$inv_dir\properties\Config.
  properties -DDirectoryProp=$inv_dir\properties\
  Directory.properties -nsac -jszenWSInv -autounload
  -Xmx128m -classpath $tmp_path;$class_path com.novell.
  zenworks.desktop.inventory.servercommon.ZENWorksInven
  toryServiceManager
```

You can start, stop, or list the services, if the Inventory Service Manager is already loaded.

- ◆ To check if the Inventory Service Manager is loaded, at the server prompt, enter **java -show**.

This will display the following message:

```
com.novell.zenworks.inventory.servercommon.ZENWorksInventoryServiceManager
```

- ◆ To start a service, enter **StartSer *service_name*** at the Inventory server prompt.
service_name refers to any of the listed services. Follow the service naming syntax when you modify the *service_name*.

For example, to start the Storer, enter **StartSer Storer**

- ◆ To stop a service, enter **StopSer *service_name*** at the Inventory server prompt.
service_name refers to any of the listed services. Follow the service naming syntax when you modify the *service_name*.

For example, to stop the Storer, enter **StopSer Storer**

- ◆ To list a service, enter **ListSer *service_name*** at the Inventory server prompt.
service_name refers to any of the listed services. Follow the service naming syntax when you modify the *service_name*.

- ◆ To list all services, enter **ListSer *** at the Inventory server prompt.

- ◆ To stop the Inventory services on the Windows NT Inventory server:

1. In the Control Panel, double-click Services.
2. Select Novell ZEN Inventory, then click Stop.

- ◆ To stop the Inventory services on the Windows 2000 Inventory server:

1. In the Control Panel, double-click Administrative Tools.
2. Double-click Services.
3. Select Novell ZEN Inventory, then click Stop.

Services on Windows NT/2000 Inventory Servers

On Windows* NT*/2000 Inventory servers, the installation program creates the Service Manager as a service. During server startup, this Inventory Service Manager is loaded as a service.

You can start, stop, or list the services, if the Inventory Service Manager (ZENworks Inventory Service) is already loaded.

To start a service:

- 1** Go to the *Installation_directory*\inv\server\wminv\bin directory.
- 2** At the prompt, enter **StartSer *service_name***.
where *service_name* refers to an Inventory service.

To stop a service:

- 1** Go to the *Installation_directory*\inv\server\wminv\bin directory.
- 2** At the prompt, enter **StopSer *service_name***.
where *service_name* refers to an Inventory service.

To stop all services (ZENworks Inventory Service), use the Windows NT/2000 services from the desktop menu.

To list a service:

- 1 Go to the `Installation_directory\inv\server\wminv\bin`.
- 2 At the prompt, enter `ListSer [-verbose] service_name`.
where `service_name` refers to an Inventory service.

Follow the service naming syntax when you modify the `service_name`.

To refer to all services, use the asterisk (*) wildcard character within double quotes `"*`. This wildcard character can be used with ListSer parameters.

Understanding the Server Configuration Service

The Server Configuration Service performs the following tasks:

- ◆ Reads the policy information from the Novell eDirectory™ and passes it to other Inventory components.
- ◆ Validates the policies to ensure that the policies are correctly configured.
- ◆ Validates the Inventory database version.

Understanding the Inventory Scanner

ZfS uses the following platform-dependent scanners to collect inventoried server hardware and software information:

- ◆ `Invnatie.nlm`, `invalid.nlm` and `mpkscan.nlm` to scan NetWare 5.1 and 6.0 inventoried servers.

The NetWare scanner collects hardware details such as: hard disk drive, BIOS, processor, bus, display adapters, network adapter cards, sound cards, memory cards, serial ports, and parallel ports.

The software scanning includes the following tasks:

- ◆ Reading `products.dat` file and scanning for the Windows applications (.exe) that are installed locally on the inventoried servers.
- ◆ Reporting the information about the scanned software like the vendor name, product name and the version number.
- ◆ `Invnatie.nlm` uses SNMP and NetWare API services to report inventory. `Invnatie.nlm` uses `nwapi.map`, `smile.map` and `suppl.map` configuration files for scanning various hardware inventory classes along with the source of information.

For example, `nwapi.map` identifies the classes that are scanned using the NetWare API services, `smile.map` identifies the classes that are scanned using SNMP services and `suppl.map` identifies the classes that are scanned using System Management BIOS (SMBIOS) services. The list of the configuration files used by `invnatie.nlm` are identified in the `hwinvsr.ini` file.

- ◆ `Invalid.nlm` reads the System Management BIOS structures and scans for certain hardware inventory such as BIOS, processors, serial and parallel ports, cache, sound adapters and display adapters.

- ◆ Mpkscan.nlm scans for the software inventory on the NetWare inventoried servers.

NOTE: The NetWare scanner that ships with Zfs 3 does not report inventory of processors for multi-processor (MPK) servers if it is not available in the SMBIOS. To scan inventory on a multiprocessor (MPK) NetWare server, you must refer to the TID 2961928 in the Knowledgebase at [Novell Technical Services \(http://support.novell.com\)](http://support.novell.com) and install the latest patch.

The NetWare scanner also does not report the virtual memory size.

- ◆ Invnativ.dll and invscan.exe to scan Windows NT/2000 inventoried servers.

The Windows scanner collects hardware details such as: floppy disk drive, hard disk drive, BIOS, bus, mouse, keyboard, display adapters, network adapter cards, modems, sound cards, memory cards, serial ports, and parallel ports. The software scanning includes checking for applications on the inventoried servers and reporting the information about the scanned software, such as the vendor name, and the product name and version.

NOTE: On Windows NT/2000 servers, the Zfs 3 Inventory scanner will not scan for Tape drives, Jaz* drivers, and Zip* drives. The Inventory scanner will not accurately report the physical memory if the memory is greater than 2 GB.

Invscan.exe leverages DMI and WMI support on the inventoried server to report hardware inventory.

The scan information collected by the scanners is stored as scan data files (.str) locally on the inventoried server. The scanned data is transferred to the Inventory server using the XML-RPC protocol. The Scan Collector receives and stores the .str files in the scan directory (scandir).

The following sections contain detailed information about the Inventory scanners:

- ◆ “How the Scanners Collect server Inventory Data” on page 731
- ◆ “Scanning Process Flowchart” on page 733
- ◆ “Software Information Collected by the Scanners” on page 733
- ◆ “DMI-Compliant Scanners” on page 734
- ◆ “WMI-Compliant Scanners” on page 735
- ◆ “SNMP-Compliant Scanners” on page 736
- ◆ “Hardware Data Collected by the Scanners” on page 736

How the Scanners Collect server Inventory Data

The scanning process is as follows:

- ◆ Subscribers must be installed and configured on the inventoried servers for Tiered Electronic Distributions (TED).
- ◆ The Server Inventory policy lets you configure the scanning schedule based on which the policy engine schedules and enforces scanning at the inventoried server.
- ◆ The Policy Enforcer triggers the Scanner, which reads the following inventory settings defined in the Inventory Service object and the Server Inventory policy.
 - ◆ **Enable Scan of Machines:** By default, the Enable Scan of Machines option is selected in the Inventory Service object property page. The Scanner collects the inventory information of the inventoried servers.
 - ◆ **Scan Directory Path:** Browse to select the DN of the Inventory Service object of the destination Inventory server.

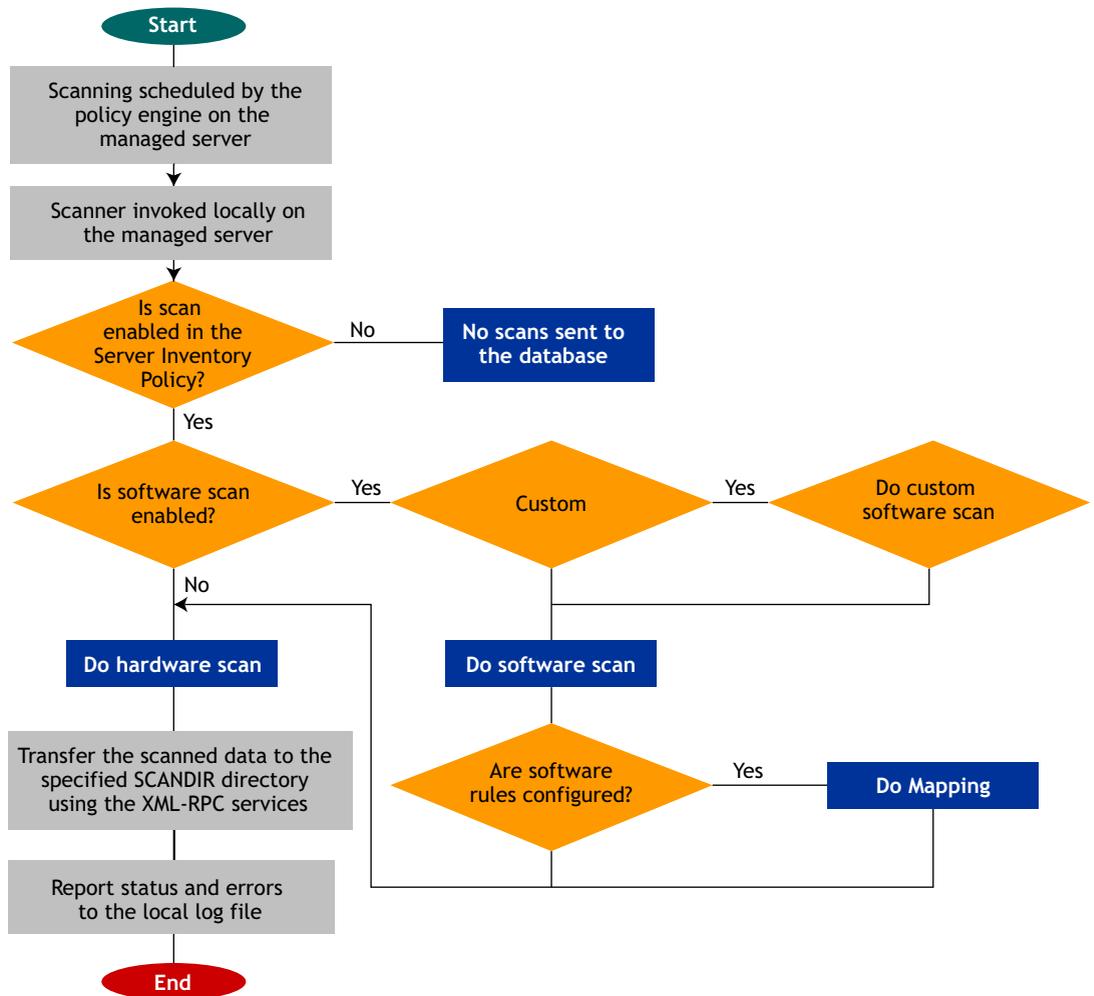
- ◆ **Enable Software Scan:** If the Enable Software Scan option is enabled in the Server Inventory policy, the Scanner collects information about software applications.
- ◆ **Custom Scan Editor:** If the Enable Software Scan option is enabled, the Scanner reports the software information. You configure the applications that you want the Scanner to collect information by using the Custom Scan Editor. You can configure the Custom Scan Editor to find information about the products that are installed on the inventoried server and do not have any information in their respective header files or the information in the header files is not accurate. The Inventory scanner picks the information configured in the Custom Scan Editor and stores it into the database
- ◆ **Configuration Editor:** If the Enable Software Scan option is enabled, you can configure the software rules to be referred while reporting the software inventory. For Windows inventoried servers, you can additionally configure for the asset information and the zip names that are to be referred during hardware inventory.
- ◆ **Enable DMI Scan:** If the server is instrumented for DMI, the scanners query the DMI Service Layer. For more information, see [“DMI-Compliant Scanners” on page 734](#)
- ◆ **Enable WMI Scan:** If the servers are WMI-compliant, the scanners also collect the hardware data by querying the WMI information. The scanners also probe the servers for hardware data. For more information, see [“DMI-Compliant Scanners” on page 734](#)

We recommend that you instrument DMI/WMI on your inventoried servers and install DMI/WMI components that are supplied by the vendors.

- ◆ The scan data of each inventoried server is stored as .str files in the scandir directory on the Inventory server. The .str file follows the filename convention: *macaddress_gmt_sequencenumber.str*, where *macaddress* is the MAC Address of the inventoried server, *gmt* is the time at which the inventoried server is scanned for the first time, and *sequencenumber* is the internal sequencing number of the inventoried server, which is always zero ('0') to indicate a full scan. For example, 00508b12b2c4_944029836000_0.str is the .str file for the server with the MAC address of 00508b12b2c4, the GMT of 944029836000, and the internal sequencing number of 0.
- ◆ The Scanner reports errors in local log files only (invagent.log and invnatve.log). The log file is stored in the sys:\etc directory on NetWare inventoried servers and in the windows directory or the temp directory on Windows NT/2000 inventoried servers.

Scanning Process Flowchart

The following flowchart illustrates the hardware and software scanning process:



Software Information Collected by the Scanners

The scanners follow this process for software scanning:

- ◆ They collect the information about the software on the inventoried servers.
- ◆ They customize the software scanning using the Custom Scan Editor.

By default, the software scanning includes collecting version information of files with .exe file extensions.

If Windows software applications (files with .exe extensions) are installed on the NetWare inventoried server, the scanner interprets the Microsoft Portable Executable (PE) format of these files to collect and report software information.

If the software applications are installed using Microsoft* Installer on the Windows NT/2000 inventoried server, the scanners use the information from Microsoft Installer (MSI). Otherwise, the scanners collect the software information from the header of the software application files.

- ◆ They report the information about the scanned software, such as name of the software product for each product version and the software vendor.

After the scan data is stored in the database, you can view or query the software information.

If you want to know the mode used by the Scanner to scan the inventoried server, see the [inventory summary of the inventoried server](#) > Hardware/Software Inventory > Software > Inventory Scanner Information > Scan Mode.

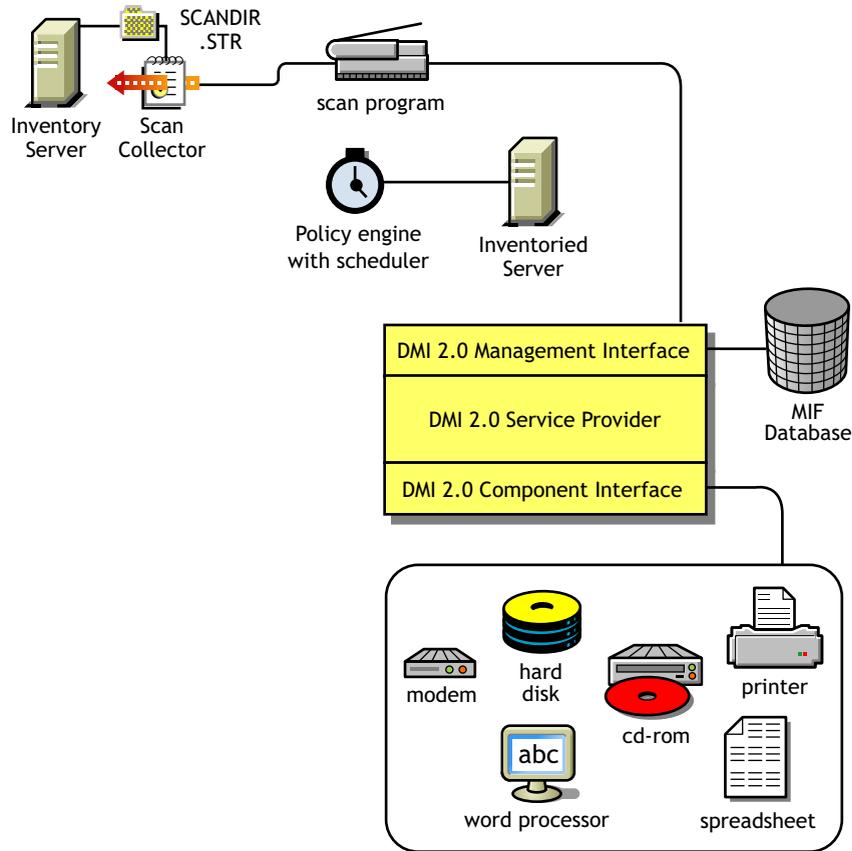
If you want to know the technology available on the inventoried server such as DMI, WMI, and others, see the [inventory summary of the inventoried server](#) > Hardware/Software Inventory > General > System Information > Management Technology.

DMI-Compliant Scanners

The scanners for scanning inventoried servers (Windows NT/2000) also include scanning based on the industry-standard Desktop Management Interface (DMI) specification 2.0. These programs use the Management Interface (MI) of DMI to look for the hardware components installed on the inventoried server. The scanners will scan for specific components that are instrumented on the inventoried server through DMI. The scanners will query the DMI service layer to retrieve this information.

The MI allows the DMI-compliant scanners to probe the Service Provider within the Service Layer. The Service Provider collects information from the manageable components and stores the collected information in the Management Information Format database. The Component Interface (CI) communicates with the manageable components and the Service layer.

The following figure shows the scanner interaction with DMI.



For more information on DMI standards, see the [DMTF Web site \(http://www.dmtf.org\)](http://www.dmtf.org).

To scan the DMI data of the inventoried servers, you need to instrument the inventoried server by installing the vendor-specific components.

For example, if you have a Compaq* Family model server running under Windows NT, download the Management Product software - Compaq Insight Management Desktop Agents software for Windows NT from the Compaq web site.

For Dell* servers, access the DM/Desktop Management Utilities software from the Dell web site.

WMI-Compliant Scanners

The scanners collect hardware data from Windows NT versions 4.0/2000 inventoried servers based on Microsoft's Windows Management Instrumentation (WMI) specification.

WMI is the Microsoft implementation of Web-Based Enterprise Management (WBEM) that enables accessing management information in an enterprise environment. WMI 1.5 is fully compliant with Common Information Model (CIM) schema, which is an industry standard. For more information, see the [Microsoft WMI Web site \(http://www.microsoft.com/hwdev/WMI\)](http://www.microsoft.com/hwdev/WMI). WMI also works with existing management standards, such as DMI and SNMP.

The scanners use WMI to look for the hardware components installed on the inventoried server. The scanners also scan for specific components that are instrumented on the inventoried server through WMI.

WMI-compliant scanners are supported on Windows NT versions 4.0 /2000 inventoried servers only.

You can view the WMI data of the inventoried servers in the Server Inventory Summary.

To obtain WMI information from the Windows NT version 4.0 inventoried server:

- 1 Download Microsoft's Windows Management Instrumentation - Core Software Installation from the [Microsoft WMI Web site \(http://msdn.microsoft.com/downloads/default.asp?url=/downloads/sample.asp?url=/msdn-files/027/001/576/msdncompositedoc.xml\)](http://msdn.microsoft.com/downloads/default.asp?url=/downloads/sample.asp?url=/msdn-files/027/001/576/msdncompositedoc.xml).

Only the WMI Core Software Installation download is required to instrument an inventoried server for WMI. To troubleshoot any WMI related problems, you can avail WMI SDK download.

IMPORTANT: On Windows 2000 servers, WMI Core Software is already installed.

- 2 Install WMI Core Software on Windows NT version 4.0 servers.

SNMP-Compliant Scanners

The Windows scanners use the Simple Network Management Protocol (SNMP) to report network adapter information and network information such as IP address, subnet mask, etc.

The NetWare scanners collect certain hardware (device) and network information based on SNMP. Additionally, the NetWare scanner also uses SNMP to report software installed and registered in products.dat. The NetWare scanner uses SNMP v2.0 and the services of hostmib.nlm, ipxrtr.nlm, and ipxrtrnm.nlm. For more information on the details about dependent MIBs and etc., see [“Hardware Data Collected by the Scanners” on page 736](#)

Hardware Data Collected by the Scanners

The Inventory Agent collects the following hardware information on the NetWare inventoried servers:

Scan Data	SNMP Details	SMBIOS Details
System.Type	SNMP v2.0 RFC1213.MIB	Not applicable
System.MachineName	SNMP v2.0 RFC1213.MIB	Not applicable
System.AssetTag	Not applicable	SMBIOS v2.3 Type 3 structure
System.Model	Not applicable	SMBIOS v2.3 Type 1 structure
System.ModelNumber	Not applicable	SMBIOS v2.3 Type 3 structure
System.SystemIdentifier	Not applicable	Not applicable
System.ManagementTechnology	Not applicable	Not applicable
System.DNName	Not applicable	Not applicable
System.TreeName	Not applicable	Not applicable

Scan Data	SNMP Details	SMBIOS Details
NetworkAdpater.MACAddress	SNMP v2.0 RFC1213.MIB	Not applicable
IP.Address	SNMP v2.0 RFC1213.MIB	Not applicable
IP.Subnet (Subnet Mask)	SNMP v2.0 RFC1213.MIB	Not applicable
NetworkAdapter.MACAddress	Not applicable	Not applicable
IPX.Adress	SNMP v2.0 IPX.MIB	Not applicable
NetworkAdapter.MACAddress	SNMP v2.0 IPX.MIB	Not applicable
DNS.HostName	Not applicable	Not applicable
NetworkAdapter.Speed	SNMP v2.0 RFC1213.MIB	Not applicable
NetworkAdapter.Name	SNMP v2.0 RFC1213.MIB	Not applicable
NetworkAdapter.PermAddress	Not applicable	Not applicable
NetworkAdapter.AdapterType	SNMP v2.0 RFC1213.MIB	Not applicable
NetworkAdapter.ProviderName	SNMP v2.0 RFC1213.MIB	Not applicable
NetworkAdapter.DriverDescription	SNMP v2.0 RFC1514.MIB	Not applicable
NetworkAdapter.DriverName	SNMP v2.0 RFC1514.MIB	Not applicable
NetworkAdapter.DriverVersion	SNMP v2.0 RFC1514.MIB	Not applicable
Zenworks_ZENNetworkAdapter---offset	SNMP v2.0 RFC1514.MIB	Not applicable
Processor.stepping	Not applicable	Not applicable
Processor.DeviceID	Not applicable	SMBIOS v2.3 Type 4 structure
Processor.Family	Not applicable	SMBIOS v2.3 Type 4 structure
Processor.OtherFamily	Not applicable	SMBIOS v2.3 Type 4 structure
Processor.MaxClockSpeed	Not applicable	SMBIOS v2.3 Type 4 structure

Scan Data	SNMP Details	SMBIOS Details
Processor.CurrentClockSpeed	Not applicable	SMBIOS v2.3 Type 4 structure
Processor.Role	Not applicable	SMBIOS v2.3 Type 4 structure
Processor.UpgradeMethod	Not applicable	SMBIOS v2.3 Type 4 structure
Processor.Description	Not applicable	SMBIOS v2.3 Type 4 structure
Processor.Name	Not applicable	SMBIOS v2.3 Type 4 structure
BIOS.Manufacturer	Not applicable	SMBIOS v2.3 Type 0 structure
BIOS.BIOSDate	Not applicable	SMBIOS v2.3 Type 0 structure
BIOS.BIOSIDBytes	Not applicable	Not applicable
BIOS.Caption	Not applicable	Not applicable
BIOS.SerialNumber	Not applicable	Not applicable
BIOS.Version	Not applicable	SMBIOS v2.3 Type 0 structure
BIOS.PrimaryBIOS	Not applicable	Not applicable
BIOS.Size	Not applicable	Not applicable
Bus.Type	SNMP v2.0 RFC1514.MIB	Not applicable
Bus.Name	Not applicable	Not applicable
Bus.Description	SNMP v2.0 RFC1514.MIB	Not applicable
Bus.Version	Not applicable	Not applicable
Monitor.NumberOfColorPlanes	Not applicable	Not applicable
Monitor.HorizontalResolution	Not applicable	Not applicable
Monitor.VerticalResolution	Not applicable	Not applicable
Monitor.DisplayType	Not applicable	Not applicable
Monitor.MemoryType	Not applicable	Not applicable
Monitor.MaxMemorySupported	Not applicable	Not applicable
Monitor.Bitsperpixel	Not applicable	Not applicable
Monitor.ControllerDescription	Not applicable	SMBIOS v2.3 Type 10 structure

Scan Data	SNMP Details	SMBIOS Details
Monitor.MaxRefreshrate	Not applicable	Not applicable
Monitor.MinRefreshrate	Not applicable	Not applicable
Mointor.DACType	Not applicable	Not applicable
Monitor.ChipSet	Not applicable	Not applicable
Monitor.ProviderName	Not applicable	Not applicable
Monitor.VideoBIOSManufacturer	Not applicable	Not applicable
Monitor.VideoBIOSVersion	Not applicable	Not applicable
Monitor.VideoBIOSReleaseDate	Not applicable	Not applicable
Monitor.VideoBIOS.IsShadowed	Not applicable	Not applicable
ParallelPort.Name	Not applicable	SMBIOS v2.3 Type 8 structure
ParallelPort.DMASupport	Not applicable	Not applicable
ParallelPort.Address	Not applicable	Not applicable
ParallelPort.IRQ	Not applicable	Not applicable
SerialPort.Name	Not applicable	Not applicable
SerialPort.Address	Not applicable	SMBIOS v2.3 Type 8 structure
SerialPort.IRQ	Not applicable	Not applicable
CDROMDrive.DeviceID(*)	Not applicable	Not applicable
CDROMDrive.Manufacture	Not applicable	Not applicable
CDROMDrive.Description	SNMP v2.0 RFC1514.MIB	Not applicable
CDROMDrive.Caption	SNMP v2.0 RFC1514.MIB	Not applicable
HardDrive.Media Type	SNMP v2.0 RFC1514.MIB	Not applicable
HardDrive.Vendor	Not applicable	Not applicable
HardDisk.Description	SNMP v2.0 RFC1514.MIB	Not applicable
HardDisk.Cylinders	Not applicable	Not applicable
HardDisk.Heads	Not applicable	Not applicable
HardDisk.Sectors	Not applicable	Not applicable
HardDisk.Capacity	SNMP v2.0 RFC1514.MIB	Not applicable

Scan Data	SNMP Details	SMBIOS Details
FileSystem.Name	Not applicable	Not applicable
InventoryScanner.Version	Not applicable	Not applicable
InventoryScanner.LastScanDate	Not applicable	Not applicable
InventoryScanner.InventoryServer	Not applicable	Not applicable
InventoryScanner.ScanMode	Not applicable	Not applicable
SoundCard.Description	Not applicable	SMBIOS v2.3 Type 10 structure
SoundCard.Name	Not applicable	Not applicable
SoundCard.Manufacturer	Not applicable	Not applicable
Cache.Level	Not applicable	Not applicable
Cache.WritePolicy	Not applicable	Not applicable
Cache.ErrorCorrection	Not applicable	SMBIOS v2.3 Type 7 structure
Cache.Type	Not applicable	SMBIOS v2.3 Type 7 structure
Cache.LineSize	Not applicable	Not applicable
Cache.ReplacementPolicy	Not applicable	Not applicable
Cache.ReadPolicy	Not applicable	Not applicable
Cache.Associativity	Not applicable	SMBIOS v2.3 Type 7 structure
Cache.Speed	Not applicable	SMBIOS v2.3 Type 7 structure
Cache.Size	Not applicable	Not applicable
UCS.DNName	Not applicable	Not applicable
UCS.PrimaryOwnerContact	Not applicable	Not applicable
UCS.PrimaryOwnerName	Not applicable	Not applicable
Slot.Description	Not applicable	SMBIOS v2.3 Type 9 structure
Slot.MaxDataWidth	Not applicable	SMBIOS v2.3 Type 9 structure
Slot.ThermalRating	Not applicable	Not applicable
LogicalDrive.Name	Not applicable	Not applicable
LogicalDrive.DeviceID	Not applicable	Not applicable
LogicalDrive.VolumeSerialNumber	Not applicable	Not applicable

Scan Data	SNMP Details	SMBIOS Details
FileSystem.Name	Not applicable	Not applicable
FileSystem.Type	Not applicable	Not applicable
FileSystem.TotalSize	Not applicable	Not applicable
FileSystem.FreeSpace	Not applicable	Not applicable
FileSystem.DeviceID	Not applicable	Not applicable
Operating System.OSType	Not applicable	Not applicable
OperatingSystem.Version	Not applicable	Not applicable
OperatingSystem.Codepage	Not applicable	Not applicable
OperatingSystem.InstallDate	Not applicable	Not applicable
OperatingSystem.SizeStoredInPagingFiles	Not applicable	Not applicable
OperatingSystem.Caption	Not applicable	Not applicable
OperatingSystem.TotalVisibleMemorySize	Not applicable	Not applicable
OperatingSystem.Role	Not applicable	Not applicable
NetWareOperatingSystem.AccountingVersion	Not applicable	Not applicable
NetWareOperatingSystem.InternetBridgeSupport	Not applicable	Not applicable
NetWareOperatingSystem.MaxNumberOfConnections	Not applicable	Not applicable
NetWareOperatingSystem.PeakConnectionsUsed	Not applicable	Not applicable
NetWareOperatingSystem.PrintServerVersion	Not applicable	Not applicable
NetWareOperatingSystem.QueueingVersion	Not applicable	Not applicable
NetWareOperatingSystem.RevisionLevel	Not applicable	Not applicable
NetWareOperatingSystem.SecurityRevisionLevel	Not applicable	Not applicable
NetWareOperatingSystem.SFTLevel	Not applicable	Not applicable
NetWareOperatingSystem.TTSLevel	Not applicable	Not applicable
NetWareOperatingSystem.VAPVersion	Not applicable	Not applicable
NetWareOperatingSystem.VirtualConsoleVersion	Not applicable	Not applicable
NetWareOperatingSystem.InternalNetworkNumber	Not applicable	Not applicable

The Inventory Agent collect the following hardware information on the Windows NT/2000 inventoried servers:

Scan Data	DMI Class and Attribute	WMI Class and Attribute
System.Type	Not applicable	Win32_SystemEnclosure.Manufacturer
System.MachineName	Not applicable	Not applicable
System.AssetTag	DMTF System Enclosure 001.2	Win32_SystemEnclosure.SMBIOSAssetTag
System.Model	Not applicable	Win32_SystemEnclosure.Model
System.ModelNumber	Not applicable	Win32_SystemEnclosure.SerialNumber
System.SystemIdentifier(GUID)	Not applicable	Not applicable
System.ManagementTechnology	Not applicable	Not applicable
System.DNNName	Not applicable	Not applicable
System.TreeName	Not applicable	Not applicable
NetworkAdpater.MACAddress	Not applicable	Win32_NetworkAdapterConfiguration.MACAddress (Only on Windows NT/2000, get through association Win32_NetworkAdapterSetting)
IP.Address	Not applicable	Win32_NetworkAdapterConfiguration.IPAddress (Only on Windows NT/2000, get through association Win32_NetworkAdapterSetting)
IP.Subnet (Subnet Mask)	Not applicable	Win32_NetworkAdapterConfiguration IPSubnet (Only on Windows NT/2000, get through association Win32_NetworkAdapterSetting)
NetworkAdapter.MACAddress	Not applicable	Win32_NetworkAdapterConfiguration.MACAddress (Only on Windows NT/2000, get through association Win32_NetworkAdapterSetting)
IPX.Adress	Not applicable	Win32_NetworkAdapterConfiguration.IPXAddress (Only on Windows NT/2000, get through association Win32_NetworkAdapterSetting)
NetworkAdapter.MACAddress	Not applicable	Win32_NetworkAdapterConfiguration.MACAddress (Only on Windows NT/2000, get through association Win32_NetworkAdapterSetting)
DNS.HostName	Not applicable	Win32_NetworkAdapterConfiguration.DNSHostName + DNSDomain (Only on Windows NT/2000, get through association Win32_NetworkAdapterSetting)
Modem.Description	Not applicable	Win32_POTSModem.Description
Modem.Name	Not applicable	Win32_POTSModem.Name

Scan Data	DMI Class and Attribute	WMI Class and Attribute
Modem.Vendor	Not applicable	Win32_POTSModem.ProviderName
Modem.DeviceID	Not applicable	Win32_POSTSModem.DeviceID
NetworkAdapter.Speed	DMTF Network Adapter 802 Port 001.5	Win32_NetworkAdapter.MaxSpeed (Only on Windows NT, when Win32_NetworkAdapter.AdapterType=Ethernet 802.3 or Fiber Distributed Data Interface (FDDI))
NetworkAdapter.Name	Not applicable	Win32_NetworkAdapter.Name (Only on Windows NT, when Win32_NetworkAdapter.AdapterType=Ethernet 802.3 or FDDI)
NetworkAdapter.PermAddress	DMTF Network Adapter 802 Port 001.2	Win32_NetworkAdapter.PermanentAddress (Only on Windows NT, when Win32_NetworkAdapter.AdapterType=Ethernet 802.3 or FDDI)
NetworkAdapter.AdapterType	Not applicable	Win32_NetworkAdapter.AdapterType (Only on Windows NT, when Win32_NetworkAdapter.AdapterType=Ethernet 802.3 or FDDI)
NetworkAdapter.ProviderName	Not applicable	Win32_NetworkAdapter.Manufacture (Only on Windows NT, when Win32_NetworkAdapter.AdapterType=Ethernet 802.3 or FDDI)
NetworkAdapter.DriverDescription	DMTF Network Adapter Driver 001.Driver Software Description	Win32_SystemDriver.Description (Only on Windows NT, when Win32_SystemDriver.Name=Win32_NetworkAdapter.ServiceName)
NetworkAdapter.DriverName	DMTF Network Adapter Driver 001.Driver Software Name	Win32_SystemDriver.PathName (Only on Windows NT, when Win32_SystemDriver.Name=Win32_NetworkAdapter.ServiceName)
NetworkAdapter.DriverVersion	DMTF Network Adapter Driver 001.Driver Software Version	Not applicable
Login.CurrentLoggedinUser	Not applicable	Not applicable
Login.DomainName	Not applicable	Win32_ComputerSystem.Domain
NWClient.Version	Not applicable	Not applicable
Processor.stepping	Not applicable	CIM_Processor.Stepping
Processor.DeviceID	Not applicable	CIM_Processor.DeviceID
Processor.Family	DMTF Processor 004.3	CIM_Processor.Family

Scan Data	DMI Class and Attribute	WMI Class and Attribute
Processor.OtherFamily	Not applicable	CIM_Processor.OtherFamilyDescription
Processor.MaxClockSpeed	DMTF Processor 004.5	CIM_Processor.MaxClockSpeed
Processor.CurrentClockSpeed	DMTF Processor 004.6	CIM_Processor.CurrentClockSpeed
Processor.Role	DMTF Processor 004.2	CIM_Processor.ProcessorType
Processor.Upgrade	DMTF Processor 004.7	CIM_Processor.UpgradeMethod
Processor.Description	Not applicable	CIM_Processor.Description
Processor.Name	Not applicable	CIM_Processor.Name
BIOS.Manufacturer	DMTF SystemBIOS 001.2	Win32_BIOS.Manufacturer
BIOS.BIOSDate	Not applicable	Win32_BIOS.InstallDate
BIOS.BIOSIDBytes	Not applicable	Not applicable
BIOS.Copyright	Not applicable	Win32_BIOS.Caption
BIOS.SerialNumber	Not applicable	Win32_BIOS.SerialNumber
BIOS.BIOSType	DMTF SystemBIOS 001.3	Win32_BIOS.SMBIOSBIOSVersion
BIOS.PrimaryBIOS	DMTF SystemBIOS 001.9	Win32_BIOS.PrimaryBIOS
BIOS.Size	DMTF SystemBIOS 001.4	Not applicable
Bus.Type	Not applicable	Win32_Bus.BusType
Bus.Name	Not applicable	Win32_Bus.Name
Bus.Description	Not applicable	Win32_Bus.Descriptipon
Bus.Version	Not applicable	Not applicable
Bus.DeviceID	Not applicable	Win32_Bus.DeviceID
IRQ.Number	DMTF IRQ 002.IRQNumber	CIM_IRQ.IRQNumber
IRQ.Availability	DMTF IRQ 002.Availability	CIM_IRQ.Availability
IRQ.TriggerType	DMTF IRQ 002.TiggerType	CIM_IRQ.TriggerType
IRQ.Shareable	DMTF IRQ 002.Shareable	CIM_IRQ.Shareable
Keyboard.Layout	DMTF Keyboard 003.Layout	CIM_Keyboard.Layout
Keyboard.Subtype	Not applicable	Not applicable
Keyboard.Type	DMTF Keyboard 003.Keybo ard.Type	CIM_Keyboard.Description
Keyboard.Fkeys	Not applicable	CIM_Keyboard.NumberOfFunctionKeys
Keyboard.Delay	Not applicable	Not applicable
Keyboard.TypematicRate	Not applicable	Not applicable

Scan Data	DMI Class and Attribute	WMI Class and Attribute
Monitor.NumberOfColorPlanes	Not applicable	Win32_VideoController.NumberOfColorPanes
Monitor.HorizontalResolution	DMTF Video 004.Current Horizontal Resolution	Win32_VideoController.CurrentHorizontalResolution
Monitor.VerticalResolution	DMTF Video 004.Current Vertical Resolution	Win32_VideoController.CurrentVerticalResolution
Monitor.DisplayType	DMTF Video 004.Video Type	Win32_VideoController.VideoArchitecture
Monitor.MemoryType	DMTF Video 004.Video Memory Type	Win32_VideoController.VideoMemoryType
Monitor.MaxMemorySupported	DMTF Video 004.Video RAM Memory Size	Win32_VideoController.MaxMemorySupported
Monitor.Bitsperpixel	DMTF Video 004.Current Number of Bits per Pixel	Win32_VideoController.CurrentBitsPerPixel
Monitor.ControllerDescription	DMTF Video 004.Video Controller Description	Win32_VideoController.Description
Monitor.MaxRefreshrate	DMTF Video 004.Maximum Refresh Rate	Win32_VideoController.MaxRefreshRate
Monitor.MinRefreshrate	DMTF Video 004.Minimum Refresh Rate	Win32_VideoController.MinRefreshRate
Monitor.DACType	Not applicable	Win32_VideoController.AdapterDACType
Monitor.ChipSet	Not applicable	Not applicable
Monitor.ProviderName	Not applicable	Not applicable
Monitor.VideoBIOSManufacturer	DMTF Video BIOS 001.BIOS Manufacturer	CIM_VideoBIOSElement.Manufacturer
Monitor.VideoBIOSVersion	DMTF Video BIOS 001.Video.BIOS Version	CIM_VideoBIOSElement.Version
Monitor.VideoBIOSReleaseDate	DMTF Video BIOS 001.Video.BIOS Release Date	CIM_VideoBIOSElement.InstallDate
Monitor.VideoBIOS.IsShadowed	DMTF Video BIOS 001.Video.Shadowing State	CIM_VideoBIOSElement.IsShadowed
ParallelPort.Name	DMTF Parallel Ports 003.Parallel Port Index	CIM_ParallelController.Name
ParallelPort.DMASupport	DMTF Parallel Ports 003.DMA Support	CIM_ParallelController.DMASupport

Scan Data	DMI Class and Attribute	WMI Class and Attribute
ParallelPort.Address	DMTF Parallel Ports 003.Parallel Base I/O Address	Not applicable
ParallelPort.IRQ	DMTF Parallel Ports 003.IRQ Used	Not applicable
SerialPort.Name	DMTF Serial Ports 004.Serial Port Index	CIM_SerialController.Name
SerialPort.Address	DMTF Serial Ports 004.Serial Base I/O Address	Not applicable
SerialPort.IRQ	DMTF Serial Ports 004.IRQ Used	Not applicable
FloppyDrive.DeviceID	DMTF Logical Drives 001.Logical Drive Name (when DMTF Logical Drives 001.Logical Drive Type=Floppy Drive(7))	Win32_LogicalDisk.DeviceID (where Win32_LogicalDisk.DriveType = 2 (Removable Disk) and Win32_LogicalDisk.MediaType = [1,10])
FloppyDrive.Manufacture	Not applicable	Not applicable
FloppyDrive.Description	DMTF Disks 003.Interface Description (when DMTF Disks 003.Storage Type=Floppy Disk(4))	Win32_LogicalDisk.Description (where Win32_LogicalDisk.DriveType = 2 (Removable Disk) and Win32_LogicalDisk.MediaType = [1,10])
FloppyDrive.MaxNumberOfCylinders	DMTF Disks 003.Number of Physical Cylinders	Not applicable
FloppyDrive.NumberOfHeads	DMTF Disks 003.Number of Physical Heads	Not applicable
FloppyDrive.SectorsPerTrack	DMTF Disks 003.Number of Physical Sectors Per Track	Not applicable
FloppyDrive.Size	DMTF Disks 003.Total Physical Size	Win32_LogicalDisk.Size (where Win32_LogicalDisk.DriveType = 2 (Removable Disk) and Win32_LogicalDisk.MediaType = [1,10])
CDROMDrive.DeviceID	DMTF Logical Drives 001.Logical Drive Name (When DMTF Logical Drives 001.Logical Drive Type = 6)	Win32_CDROMDrive.Drive
CDROMDrive.Manufacture	Not applicable	Win32_CDROMDrive.Manufacturer
CDROMDrive.Description	Not applicable	Win32_CDROMDrive.Description

Scan Data	DMI Class and Attribute	WMI Class and Attribute
CDROMDrive.Caption	Hard code: Cdrom Device (when DMTF Disks 001.Logical Drive Type = 6)	Win32_CDROMDrive.Caption
HardDrive.Media Type	DMTF Disks 003.Removable Media	Win32_DiskDrive.MediaType
HardDrive.Vendor	Not applicable	Win32_DiskDrive.Manufacturer
HardDisk.Description	DMTF Disks 003.Interface Description (when DMTF Disks 003.Storage Type=Hard Disk(3))	Win32_DiskDrive.Description
HardDisk.Cylinders	DMTF Disks 003.Number of Physical Cylinders	Win32_DiskDrive.TotalCylinders
HardDisk.Heads	DMTF Disks 003.Number of Physical Heads	Win32_DiskDrive.TotalHeads
HardDisk.Sectors	DMTF Disks 003.Number of Physical Sectors per Track	Win32_DiskDrive.SectorsPerTrack
HardDisk.Capacity	DMTF Disks 003.Total Physical Size	Win32_DiskDrive.Size
LogicalDrive.Name	Not applicable	Win32_LogicalDiskDeviceID (when Win32_LogicalDisk.DriveType = 3 (Local Disk))
LogicalDrive.VolumeSerialNumber	Not applicable	Win32_LogicalDisk.VolumeSerialNumber (when Win32_LogicalDisk.DriveType = 3 (Local Disk))
LogicalDrive.Volume (Volume Label)	Not applicable	Win32_LogicalDisk.VolumeName (when Win32_LogicalDisk.DriveType = 3 (Local Disk))
FileSystem.Drive	Not applicable	Win32_LogicalDiskDeviceID (when Win32_LogicalDisk.DriveType = 3 (Local Disk))
FileSystem.FileSystemSize	Not applicable	Win32_LogicalDisk.Size (when Win32_LogicalDisk.DriveType = 3 (Local Disk))
FileSystem.AvailableSpace	Not applicable	Win32_LogicalDisk.FreeSpace (when Win32_LogicalDisk.DriveType = 3 (Local Disk))
FileSystem.FileSystem	Not applicable	Win32_LogicalDisk.FileSystem (when Win32_LogicalDisk.DriveType = 3 (Local Disk))
OperatingSystem.OSType	Not applicable	Win32_OperatingSystem.OSType
OperatingSystem.Version	Not applicable	Win32_OperatingSystem.Version

Scan Data	DMI Class and Attribute	WMI Class and Attribute
OperatingSystem.Codepage	Not applicable	Win32_OperatingSystem.CodeSet
OperatingSystem.InstallDate	Not applicable	Win32_OperatingSystem.InstallDate
OperatingSystem.TotalSwapSpaceSize	Not applicable	Not applicable
OperatingSystem.Description	Not applicable	Win32_OperatingSystem.Caption
OperatingSystem.OtherTypeDescription	Not applicable	Win32_OperatingSystem.OtherTypeDescription
OperatingSystem.VirtualMemorySize	DMTF System Memory Settings Total Virtual Memory	Win32_OperatingSystem.TotalVirtualMemory
OperatingSystem.VisibleMemorySize	Not applicable	Win32_OperatingSystem.TotalVisibleMemorySize
OperatingSystem.Role	Not applicable	Not applicable
InventoryScanner.Version	Not applicable	Not applicable
InventoryScanner.LastScanDate	Not applicable	Not applicable
InventoryScanner.InventoryServer	Not applicable	Not applicable
InventoryScanner.ScanMode	Not applicable	Not applicable
SoundCard.Description	Not applicable	Win32_SoundDevice.Description
SoundCard.Name	Not applicable	Win32_SoundDevice.Name
SoundCard.Manufacturer	Not applicable	Win32_SoundDevice.Manufacturer
Cache.Level	DMTF System Cache 003.System Cache Level	Win32_CacheMemory.Level
Cache.WritePolicy	DMTF System Cache 003.System Cache Write Policy	Win32_CacheMemory.WritePolicy
Cache.ErrorCorrection	DMTF System Cache 003.System Cache Error Correction	Win32_CacheMemory.ErrorMethodology
Cache.Type	DMTF System Cache 003.System Cache Type	Win32_CacheMemory.CacheType
Cache.LineSize	DMTF System Cache 003.Line Size	Win32_CacheMemory.LineSize
Cache.ReplacementPolicy	DMTF System Cache 003.Replacement Policy	Win32_CacheMemory.ReplacementPolicy
Cache.ReadPolicy	DMTF System Cache 003.Read Policy	Win32_CacheMemory.ReadPolicy

Scan Data	DMI Class and Attribute	WMI Class and Attribute
Cache.Associativity	DMTF System Cache 003.Associativity	Win32_CacheMemory.Associativity
Cache.Speed	DMTF System Cache 003.System Cache Speed	Win32_CacheMemory.CacheSpeed
Cache.Size	DMTF System Cache 003.System Cache Size	Win32_CacheMemory.MaxCacheSize
MotherBoard.Version	Not applicable	Win32_BaseBoard.Version
MotherBoard.Description	Not applicable	Win32_BaseBoard.Description
MotherBoard.Slots	DMTF Motherboard 001.Nu mber of Expansion slots	Not applicable
MotherBoard.Manufacture	Not applicable	Win32_BaseBoard.Manufacture
Battery.Name	DMTF Portable Battery 002.Portable Battery Device Name	Win32_Battery.Name
Battery.Chemistry	DMTF Portable Battery 002.Portable Battery Device Chemistry	Win32_Battery.Chemistry
Battery.Capacity	DMTF Portable Battery 002.Portable Battery Design Capacity	Win32_Battery.DesignCapacity
Battery.Voltage	DMTF Portable Battery 002.Portable Battery Design Voltage	Win32_Battery.DesignVoltage
Battery.Version	DMTF Portable Battery 002.Portable Battery Smart Battery Version	Win32_Battery.SmartBatteryVersion
Battery.Manufacturer	DMTF Portable Battery 002.Portable Battery Manufacturer	Win32_PortableBattery.Manufacturer
Battery.ManufactureDate	DMTF Portable Battery 002.Portable Battery Manufacturer Date	Win32_Battery.InstallDate
Battery.SerialNumber	DMTF Portable Battery 002.Portable Battery Serial Number	Not applicable
PowerSupply.InputVoltageDescription	DMTF Power Supply 002.Power Supply Input Voltage Capability Description	CIM_UninterruptiblePowerSupply.Description

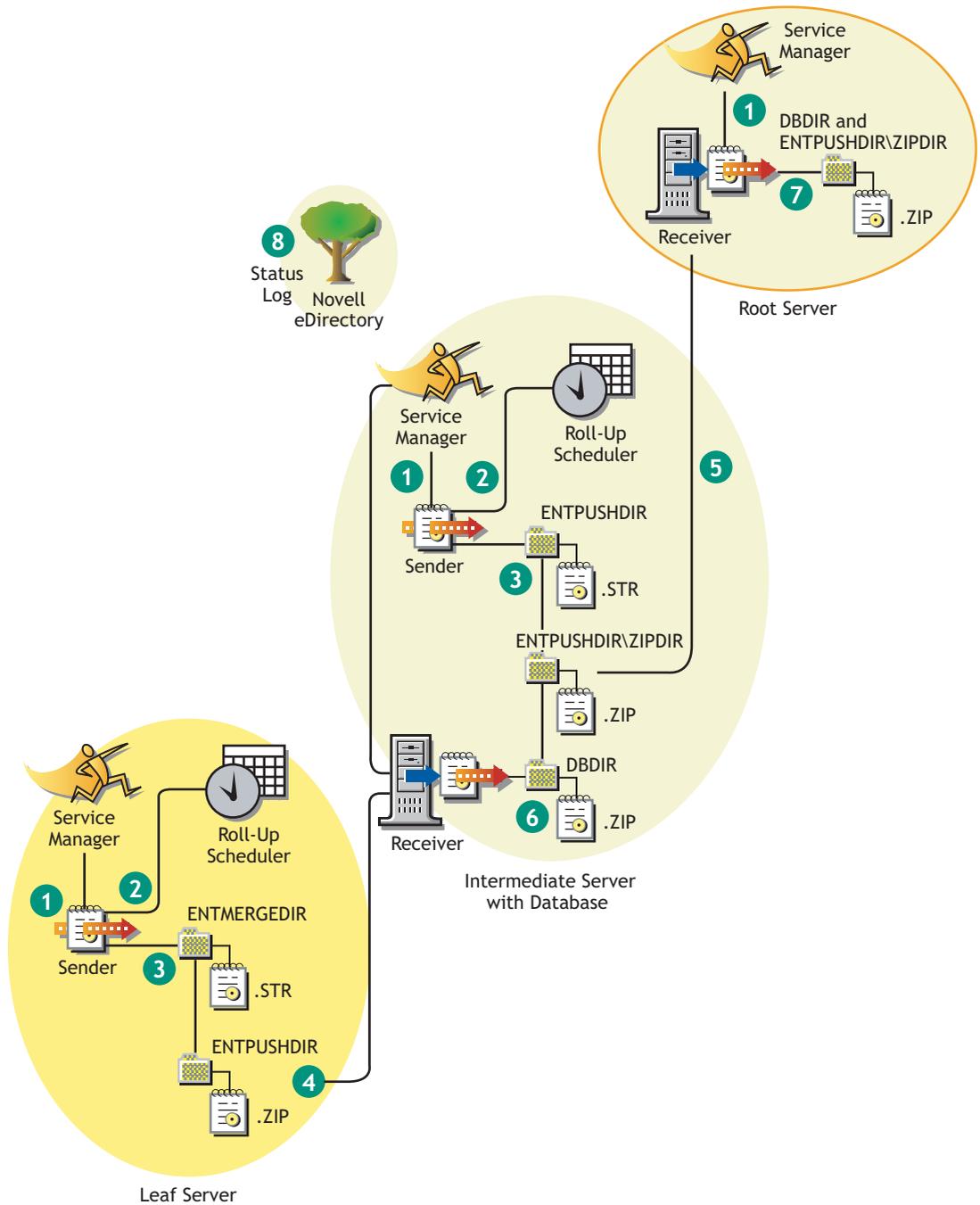
Scan Data	DMI Class and Attribute	WMI Class and Attribute
PowerSupply.Power	DMTF Power Supply 002.Total Output Power	CIM_UninterruptiblePowerSupply.TotalOutputPower
DMA.Number	DMTF DMA 001.DMA Number	CIM_DMA.DMAChannel
DMA.Description	DMTF DMA 001.DMA Description	CIM_DMA.Description
DMA.Availability	DMTF DMA 001.DMA Channel Availability	CIM_DMA.Availability
DMA_BurstMode	DMTF DMA 001.DMA BurstMode	CIM_DMA.BurstMode
UCS.DNNName	Not applicable	Not applicable
UCS.PrimaryOwnerContact	DMTF General Information 001.3	CIM_UnitaryComputerSystem.PrimaryOwnerContact
UCS.PrimaryOwnerName	DMTF General Information 001.4	CIM_UnitaryComputerSystem.PrimaryOwnerName
Pointing Device.DeviceType	Not applicable	CIM_PointingDevice.PointingType
Pointing Device.Type	DMTF Pointing Device Pointing Device Interface	CIM_PointingDevice.Name
Pointing Device.NumberOfButtons	DMTF Pointing Device Number Of Buttons	CIM_PointingDevice.NumberOfButtons
Pointing Device.DriverName	DMTF Pointing Device Pointing Device Driver Name	CIM_PointingDevice.Name
Pointing Device.DriverVersion	DMTF Pointing Device Pointing Device Driver Version	Not applicable
Pointing Device.IRQ	DMTF Pointing Device Pointing Device IRQ	Not applicable
Slot.Description	DMTF System Slots 003.Description	Not applicable
Slot.MaxDataWidth	DMTF System Slots 003.MaxDataWidth	Not applicable
Slot.ThermalRating	DMTF System Slots 003.Slot Thermal Rating	Not applicable

Understanding the Sender-Receiver

The Sender and the Receiver on the Inventory servers transfer the scan files from the lower-level Inventory servers to the higher-level Inventory servers. The Scan Collector uses the ZEN Web Server to process the XML-RPC requests. The following sections contain more information:

- ◆ [“Understanding the Sender” on page 753](#)
- ◆ [“Understanding the Receiver” on page 754](#)
- ◆ [“Understanding the Compressed Scan Data File” on page 754](#)
- ◆ [“Sender-Receiver Directories” on page 755](#)

The following illustration depicts the processing done by the Sender-Receiver.



The processing done by the Sender-Receiver is as follows:

1. The Service Manager starts the Sender-Receiver component.
2. The Roll-Up Scheduler activates the Sender at the specified roll-up time.
3. The Sender moves the scan data files (.str) from the enterprise merge directory (entmergedir) to the enterprise push directory (entpushdir) and compresses the files as a .zip file.
4. Each .zip file is again compressed with the .PRP file into a .zip file.
5. The Sender sends the .zip file from the entpushdir directory to the Receiver on the next-level Inventory server.

6. The Receiver places the .zip files to the entpushdir\zipdir directory.
7. The Receiver copies the .zip files to the entpushdir directory and deletes the .zip files from the entpushdir\zipdir directory.
8. The Receiver copies the .zip files to the database directory (dbdir) if a database is attached to the Inventory server.
9. The Sender-Receiver logs the status in eDirectory.

Understanding the Sender

The Sender is a Java* component that runs on any Leaf Server or on the Intermediate Server. The Sender is a service loaded by the Service Manager. See [“An Overview of the Inventory Components on the Inventory Server” on page 760](#) for a quick reference table of Inventory server components.

The flow of information from the Sender in the roll-up of scan data is as follows:

1. The Service Manager starts the Sender on the Inventory server. At the specified time scheduled in the Roll-Up Schedule, the Sender moves the scan data files (.str) from the enterprise merge directory (entmergedir) to the enterprise push directory (entpushdir).

The Sender compresses these .str files in the entpushdir directory of the Inventory server as a .zip file and then deletes the .str files. This .zip file is again compressed with the .PRP file into a .zip file. For more information, see [“Understanding the Compressed Scan Data File” on page 754](#).
2. The Sender creates a new record in the zeninvRollUpLog attribute of the Inventory Service object (ZenInvservice) in eDirectory with the following details: server on which the Sender compresses the .str files and the name and size of the .zip file.
3. Based on the Discard Scan Data Time in the Inventory Service object properties of the Receiver, the Sender deletes the compressed .zip files in the entpushdir directory that have been created earlier than the specified discard scan data time. This removes unwanted scan information being sent in the roll-up.
4. The Sender sends the compressed .zip files to the Receiver, with the oldest compressed files sent first.
5. The Sender after transferring the .zip file, deletes the compressed files in the entpushdir directory.
6. After the roll-up of data, the Sender updates the zeninvRollUpLog attribute of the Inventory server on which the compressed file was created with the following details: Inventory server from which the Sender transmitted the file, name of the .zip file, time of transmission, total time taken to transmit the files, and the Inventory server to which it was sent.

In case of rolling up scan data across trees, the roll-up status messages are logged into the first inventory server receiving the .zip file in the tree.

The status information for all actions of the Sender is logged in the Roll-Up Log and Server Status log. For more information, see [“Monitoring Server Inventory Using Status Logs” on page 843](#).

If the Sender is unable to connect to the Receiver, the Sender retries to connect after 10 seconds. The time interval increases exponentially by a factor of 2. After 14 retries, the Sender stops trying to connect to the Receiver. The Sender retries for approximately 23 hours before it discontinues trying. The Sender does not process any other data while it is establishing the connection.

Understanding the Receiver

The Receiver is a Java component that runs on the Intermediate Server or on the Root Server. The Receiver is a service loaded by the Service Manager. See [“An Overview of the Inventory Components on the Inventory Server” on page 760](#) for a quick reference table of Inventory server components.

The processing done by the Receiver is as follows:

1. The Receiver receives the scan .zip file from the Sender and places the file in the entpushdir\zipdir directory.
2. The Receiver copies the .zip file to the entpushdir directory and deletes the .zip files from the entpushdir\zipdir directory.

On an Intermediate Server, the file is placed in entpushdir. On an Intermediate Server with Database, or an Intermediate Server with Database and Inventoried Servers, the file is placed in entpushdir and copied to the database directory (dbdir).

3. The Receiver on the Root Server or the Root Server with Inventoried Servers receives the .zip files from the Senders and places the .zip files in the entpushdir\zipdir directory. It copies the files to the dbdir directory on the Inventory server.
4. The Receiver logs the status information in the Roll-Up log. For more information, see [“Monitoring Server Inventory Using Status Logs” on page 843](#).

Understanding the Compressed Scan Data File

The Sender compresses the scan data files (.str) into a .zip file. This .zip file is again compressed with the .PRP file into a .zip file. The .zip file (containing the .zip files and .prp) is named using the following naming conventions:

scheduledtime_inventoryservername_treename_storedstatus.zip

where *scheduledtime* refers to the date and time when the .zip file is created, *inventoryservername* refers to the Inventory server on which the .zip file was compressed, *treename* refers to the unique tree name in which the .zip file is currently located, *storedstatus* refers to the storage status of the .zip file, and *ZIP* is the file extension for the compressed files.

The *storedstatus* is represented by 0, 1, or 2. 0 indicates the .zip file has not yet been stored. 1 indicates the .zip file will be stored for the first time in the Inventory server. 2 indicates the .zip file has already been stored once.

The .zip filename changes depending on if the database is attached to the Inventory server.

The .zip file contains the .zip files and a property file. The property file is named using the following conventions:

scheduledtime_inventoryservername.prp

The property file contains the scheduled time, Inventory server name, and signature. The signature helps to authenticate the .zip file.

Each .zip file can contain a maximum of 1,000 .str files.

Sender-Receiver Directories

The following table provides a quick reference of the directories that the Sender-Receiver uses:

Server	Sender	Receiver	ENTMERGDIR	ENTPUSHDIR \ ZIPDIR	ENTPUSHDIR	DBDIR
Leaf Server, Leaf Server with Database	Runs on this Inventory server	--	Sender moves the .str files to entpushdir.	--	Sender compresses the .str files as a .zip file. Sender deletes the .str files. Sends the .zip file to the next-level Inventory server.	--
Intermediate Server	Runs on this Inventory server	Runs on this Inventory server	--	Receiver receives the .zip files from the lower-level Inventory server in this directory.	Receiver copies the .zip files from the lower-level Inventory server in this directory. Sender sends the .zip files to the next-level Inventory server.	--
Intermediate Server with Inventoried Servers	Runs on this Inventory server	Runs on this Inventory server	Sender moves the .str files to entpushdir.	Receiver receives the .zip files from the lower-level Inventory server in this directory.	Receiver copies the .zip files from zipdir into this directory. Sender sends the .zip files to the next-level Inventory server. Sender compresses the .str files in to .zip files. Sender deletes the .str files.	--
Intermediate Server with Database	Runs on this Inventory server	Runs on this Inventory server	--	Receiver receives the .zip files from the lower-level Inventory server in this directory.	Receiver copies the .zip files from ZIPDIR into this directory. Sender sends the .zip file to the next-level Inventory server.	Receiver copies the file in this directory.
Intermediate Server with Database and Inventoried Servers	Runs on this Inventory server	Runs on this Inventory server	Sender moves the .str files to entpushdir.	Receiver receives the .zip files from the lower-level Inventory server in this directory.	Receiver copies the .zip files from ZIPDIR into this directory. Sender compresses the .str files as a .zip file. Sender deletes the .str files. Sender sends the .zip file to the next-level Inventory server.	Receiver copies the file in this directory.
Root Server, Root Server with Inventoried Servers	--	Runs on this Inventory server	--	Receiver receives the .zip files from the lower-level Inventory server in this directory.	--	Receiver copies the .zip files from the lower-level Inventory server in this directory.

On the Standalone Server, the Receiver is not loaded.

Understanding the Selector

The Selector is a Java component on the Inventory server that receives the scan data from the Inventory servers. These Inventory servers can be any of the following: Leaf Server, Leaf Server with Database, Intermediate Server with Database and Inventoried Servers, Intermediate Server with Inventoried Servers, Root Server with Inventoried Servers, and Standalone Server. See [“An Overview of the Inventory Components on the Inventory Server” on page 760](#) for a quick reference table of Inventory server components.

The processing done by the Selector is as follows:

1. While scanning the inventoried server, the Scanner creates a scan data file (.str) in the scan directory (scandir) at the Inventory server for each scan done on the inventoried server. The location of scandir is obtained from the Inventory Service object. The Selector processes the .str files placed by the Scan Collector in the scandir directory.
2. The Selector checks for the following conditions to ensure that the .str file generated by the Scanner is valid:
 - ♦ The integer value that is generated by using the .str file and logged into the .str file by the Scanner and the integer value generated by using the .str file by the Selector should be the same.
 - ♦ The actual size of the .str file should be in sync with the size recorded in the .str file.

The Selector processes only valid .str files. If invalid files are present in the directory, the Selector deletes the files.

3. Based on the role of the Inventory server, the Selector copies the .str files to the dbdir directory (if the database is attached) and the entmerge directory. If the .str file already exists in the directory, it overwrites the file.

The following table lists the directories that the Selector copies or renames the files to:

Server	Copies the .str file to the Database Directory (dbdir)	Renames the .str file in the Database Directory (dbdir)	Renames the .str file in the Enterprise Merge Directory (entmergedir)
Leaf Server with Database	Yes	--	Yes
Leaf Server	--	--	Yes
Intermediate Server with Database and Inventoried Servers	Yes	--	Yes
Standalone Server	--	Yes	--
Root Server with Inventoried Servers	--	Yes	--

4. The Selector logs the status in the Server log. For more information, see [“Monitoring Server Inventory Using Status Logs” on page 843](#).

Understanding the Storer

The Storer is a Java component on the Inventory server that has a database attached to it. These Inventory servers can be any of the following: Leaf Server with Database, Intermediate Server with Database, Intermediate Server with Database and Inventoried Servers, Root Server, and Root Server with Inventoried Servers. See [“An Overview of the Inventory Components on the Inventory Server” on page 760](#) for a quick reference table of Inventory server components.

The Storer runs as a Service loaded by the Service Manager. It processes the files in the DBDIR directory.

The processing done by the Storer is as follows:

1. The Storer reads the Startup configuration parameters from the Inventory server Configuration Service.
2. From the Inventory server configuration information stored in eDirectory, the Storer looks in the database directory (dbdir) for the scan files. The Inventory server configuration information determines the location of dbdir and the database server from the eDirectory policy. The Selector places the .str files in dbdir and the Receiver places the .zip files in dbdir.
3. The Storer processes the .str files and the .zip files alternately.
4. The Storer extracts the .zip file containing the compressed .str files and the .prp file to a temp directory (dbdir/temp) and updates the database with the inventory information of the .str files for the inventoried servers.
5. The Storer updates the status in the Inventory server Status log and updates the Roll-Up log. You can view the Inventory server status information in the Inventory server Status log.

In case of rolling up scan data across trees, the roll-up status messages are logged into the first inventory server receiving the .zip file in the tree. For more information, see [“Monitoring Server Inventory Using Status Logs” on page 843](#).

Understanding the Inventory Removal Service

The Inventory Removal service is a manual service that runs on the Inventory server. You can remove the unwanted, redundant, or obsolete inventoried servers from the Inventory database using the Inventory Removal service. The Inventory Removal service removes the inventoried servers from the Inventory database through using the inventoryremovallist.txt file. To understand the Inventory Removal Service synchronization, see [“Using the Inventory Removal Service for Synchronization” on page 758](#).

The inventoryremovallist.txt file contains a list of inventoried servers that have to be removed from the Inventory database.

IMPORTANT: You can run the Inventory Removal service on the Intermediate Server only if the Intermediate Server has either inventoried servers or database attached to it.

To remove the inventoried servers from the Inventory database:

- 1 Using a text editor, create a file with the name inventoryremovallist.txt with the following contents:

```
;  
; Enter comments, if any  
  
DN or name of the inventoried server (as stored in the Inventory database)  
to be removed from the Inventory database  
DN or name of the inventoried server (as stored in the Inventory database)
```

to be removed from the Inventory database
...
...
DN or name of the *inventoried server* (as stored in the Inventory database)
to be removed from the Inventory database

A sample file is as follows:

```
CN=INT-SERVER-NDS.OU=Leaf.O=XYZ.T=XYZ-TREEzen-server.xyz.com  
CN=ROOT-SERVER-NDS.O=XYZ.T=XYZ-TREE
```

To generate the list of inventoried servers that must be removed you can either perform a query on a selected criteria or manually enter the names of the inventoried servers. For more information on Query, see [“Forming the Query and Setting the Filter Conditions” on page 839](#).

- 2** Copy the `inventoryremovallist.txt` file to the `ZENworks_installation_path\zenworks\inv\server\wminv\properties` directory.
- 3** In the `ZENworks_installation_path\zenworks\inv\server\wminv\properties\inventoryremoval.properties` file, ensure that the value of `FilePath` is the location of `inventoryremovallist.txt` (specified in Step 2).
NOTE: Ensure that the path separator is a forward slash (/) and not a backslash (\).
- 4** At the server console prompt, enter **StartSer RemoveInventory** to start the Inventory Removal service.

The processing done by Inventory Removal service is as follows:

- 1** The Inventory Removal service reads each line of the `inventoryremovallist.txt` file and creates a delete str file for each inventoried server that is listed in the `inventoryremovallist.txt` file.
The delete str file is saved in the `scandir` directory if the Selector is running, else it will be placed in the `dbdir` or `entmergedir` directories depending on the Inventory server role.
- 2** The Selector validates the delete str file and copies it into the `dbdir` and `entmergedir` directories.
- 3** The Storer reads the delete str file from `dbdir` and deletes the inventoried server from the attached Inventory database.
- 4** If the inventory deployment rolls up scan data, the delete str is also rolled up to the next level Inventory server.

The inventoried server is deleted from the Inventory database at all Inventory servers deployed at the enterprise level.

Using the Inventory Removal Service for Synchronization

The Inventory Removal Service automatically removes inventoried servers from the Inventory database when the corresponding server objects are removed from eDirectory.

Sometimes it is possible that the inventoried servers in eDirectory and in the Inventory database are not synchronized because of one or both of the following reasons:

- ◆ If you kill the Inventory Service Manager, remove some server objects in the eDirectory, and restart the Inventory Service Manager.
- ◆ If you restart a previous version of the Inventory database containing servers that have already been deleted from eDirectory.

If this happened, you can use the Inventory Removal Service to remove unwanted inventoried servers from the Inventory database so the database is once again synchronized with eDirectory.

If you know the fully qualified DN names of the inventoried servers, you can specify the DN names of these servers in the `inventoryremovallist.txt` file.

To find the server objects that were removed from eDirectory:

1. Export the list of server objects attached to the given Inventory server using an eDirectory tool like NDSREPAIR. The eDirectory tools can be downloaded from the [Cool Solutions Web site](http://www.novell.com/coolsolutions/freetools.html) (<http://www.novell.com/coolsolutions/freetools.html>).

2. To export all the server objects into a `.csv` file, use the Data Export Wizard.

NOTE: While exporting all the inventoried servers to a `.csv` files, it is necessary to select the attributes.

The exported `.csv` file will contain the DNS name and the selected attributes of the inventoried servers. However, you must remove attribute values and the double-quote characters from the `.csv` file.

3. Compare the eDirectory exported file and the `.csv` file using the file comparison utility to identify the inventoried servers that do not match the `.csv` file.

NOTE: The eDirectory output file and the `.csv` file should be in the same format for the comparison to succeed.

4. After identifying the inventoried servers that are not synchronized, place the DN names of these servers in the `inventoryremovallist.txt` file for the Inventory Removal Service to pick them up.

Understanding the Upgrade Service

The Upgrade service runs as a service loaded by the Service Manager.

The Upgrade service corrects the Inventory database schema and data to make it compatible with ZfS 3 SP1 and ZENworks for Desktops 4. The Upgrade service performs all the functions in a state-driven method. This is to make sure that the Upgrade service does not execute the same steps when one step is executed successfully. The Upgrade service runs as an uninterrupted service. Therefore, you cannot manually stop the Upgrade service. The Upgrade service stops automatically after completing all its functions.

The Database migration activity is additionally traced into a migration log, which could be found in the `installation_path\zenworks\inv\server\wminv\logs\migrationlogs` directory.

An Overview of the Inventory Components on the Inventory Server

Depending on the type of the Inventory server, the following inventory components exist on the Inventory server.

Server Component	Root Server	Root Server with Inventoried Servers	Leaf Server	Leaf Server with Database	Intermediate Server	Intermediate Server with Database and Inventoried Servers	Intermediate Server with Database	Intermediate Server with Inventoried Servers	Stand alone Server
Service Manager	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Scan Collector	--	Yes	Yes	Yes	--	Yes	--	Yes	Yes
Selector	--	Yes	Yes	Yes	--	Yes	--	Yes	Yes
Storer	Yes	Yes	--	Yes	--	Yes	Yes	--	Yes
Sender	--	--	Yes	Yes	Yes	Yes	Yes	Yes	--
Receiver	Yes	Yes	--	--	Yes	Yes	Yes	Yes	--
Database	Yes	Yes	--	Yes	--	Yes	Yes	--	Yes
Inventory Removal Service	Yes	Yes	Yes	Yes	--	Yes	Yes	Yes	Yes
Upgrade Service	Yes	Yes	--	Yes	--	Yes	Yes	--	Yes

Understanding the Inventory Database

ZfS Server Inventory provides a centralized Common Information Model (CIM)-compliant Sybase database. The Inventory database serves as a repository of hardware and software information for the servers. The network administrator can view the inventory information, query the database, and generate inventory reports in ConsoleOne. For more information, see [Chapter 28, “Understanding the ZENworks for Servers Inventory Database Schema,” on page 787](#)

Understanding ZfS Inventory Attributes

The following table lists the Server Inventory attributes that ZENworks for Servers uses.

Each row in the table has:

- ◆ Name of the attribute as displayed in the Inventory Database Export Wizard in ConsoleOne
- ◆ Name of the attribute in the exported .CSV file (first row in the .CSV file)
- ◆ Inventory database attribute name
- ◆ Type of the attribute in the Inventory database
- ◆ Length of the attribute in the Inventory database
- ◆ Brief description of the attribute

Hardware and software enumerated values are listed separately, following the table.

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
General-NDSName-Label	NDSName_LABEL	ManageWise.NDSName.La bel	String	254	The DN name of the inventoried server registered in eDirectory
SystemInfo.Description	Asset_Description	Zenworks.SystemInfo.Desc ription	String	254	Description of the system asset information
SystemInfo.Caption	Asset_Caption	Zenworks.SystemInfo.Capti on	String	64	Identifying information of the computer
SystemInfo.Tag	Asset_Asset Tag	Zenworks.SystemInfo.Tag	String	254	Asset tag number that the ROM-based setup program creates. This is unique to every inventoried server.
SystemInfo.ModelNumber	Asset_Model Number	Zenworks.SystemInfo.Mode l	String	64	Model number value for the computer, assigned during manufacture
SystemInfo.SerialNumber	Asset_Serial Number	Zenworks.SystemInfo.Serial Number	String	64	Model serial number value for the computer, assigned during manufacture
SystemInfo.ManagementTec hnology	Asset_Management Technology	Zenworks.SystemInfo.Mana gementTechnology	Integer		The management technology available on the computer system
CurrentLoginUser.Name	Current Login User.Name	ManageWise."User".Name	String	254	User logged in to the Primary eDirectory tree when the inventoried server was scanned
LastLoginUser.Name	Last Login User.Name	ManageWise."User".Name	String	254	User logged in last to the Primary eDirectory tree when the inventoried server was scanned
Product.Name	Applications_Name	CIM.Product.Name	String	254	Name of the software application

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
Product.Vendor	Applications_Vendor	CIM.Product.Vendor	String	254	Name of the software application manufacturer
Product.Version	Applications_Version	CIM.Product.Version	String	64	Version of the software application
Product.Location	Applications_Path	CIM.Directory.Location	String	254	The product installation path
Product.IdentifyingNumber	Applications_Identifying Number	CIM.Product.IdentifyingNumber	String	64	Microsoft product ID
WinOperating System.OSType	Windows_Name	ZENworks.WINOperatingSystem.OSType	Unsigned Small Integer (enum)		Operating system name. For example, Windows NT/Windows 2000. See “Enumeration Values for SOFTWARE-Operating Systems-Name” on page 782.
WinOperating System.Version	Windows_Version	ZENworks.WINOperatingSystem.Version	String	254	Version of the operating system
WinOperating System.Caption	Windows_Caption	ZENworks.WINOperatingSystem.Caption	String	64	Short name of the operating system. For example, Windows NT
WinOperating System.Role	Windows_Role	ZENworks.WINOperatingSystem.Role	Integer (enum)		The role of the computer system. For example, server or workstation
WinOperating System.OtherTypeDescription	Windows_Other Description	ZENworks.WINOperatingSystem.Description	String	254	More description about the operating system
WinOperating System.InstallDate	Windows_Install Date	ZENworks.ZENOperatingSystem.InstallDate	String	25	Install date of the operating system
WinOperating System.CodePage	Windows_Code Page	ZENworks.WINOperatingSystem.CodePage	String	254	Current language code page being used
WinOperating System.TotalVisibleMemorySize	Windows_Total Memory (MB)	ZENworks.WINOperatingSystem.TotalVisibleMemorySize	Integer		Total memory as reported by the Windows operating system
WinOperating System.TotalVirtualMemorySize	Windows_Total Virtual Memory (MB)	ZENworks.WINOperatingSystem.TotalVirtualMemorySize	Integer		Total virtual memory as reported by the Windows operating system

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
InventoryScanner.Version	Scanner Information_Version	ZENworks.InventoryScanner.Version	String	64	Version of the scanner running on the inventoried server
InventoryScanner.LastScanDate	Scanner Information_Last Scan Date	ZENworks.InventoryScanner.LastScanDate	Unsigned Integer		The date when the scanner was last scanned. Stored as milliseconds time value so that it could be read and displayed in any an appropriate date format
InventoryScanner.Inventory Server	Scanner Information_Inventory Server	ZENworks.InventoryScanner.InventoryServer	String	254	Name of the inventory server to which the scans are sent. It is not the complete DN of the server name
InventoryScanner.ScanMode	Scanner Information_Scan Mode	ZENworks.InventoryScanner.ScanMode	Integer (enum)		The management technology used by the scanner, such as WMI or DMI, for scanning the computer system
NetWareClient.Version	Netware Client_Version	ZENworks.NetWareClient.Version	String	64	Version of the NetWare client software installed on the inventoried server
NetworkAdapterDriver.Description	Network Adapter Driver_Description	ZENworks.NetworkAdapterDriver.Description	String	254	Description of the network adapter driver installed on the inventoried server. For example, IBM 10/100 Ethernet adapter, EN-2420Px Ethernet adapter
NetworkAdapterDriver.Name	Network Adapter Driver_Name	ZENworks.NetworkAdapterDriver.Name	String	254	Name of the network adapter driver software installed that corresponds to the adapter. For example, ne2000.sys, pppmac.vxd, and others

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
NetworkAdapterDriver.Version	Network Adapter Driver_Version	ZENworks.NetworkAdapterDriver.Version	String	64	Network adapter driver version
PointingDeviceDeviceDriver.Name	Pointing Device Driver_Name	ZENworks.PointingDeviceDeviceDriver.Name	String	254	Name of the mouse driver installed on the inventoried server
PointingDeviceDeviceDriver.Version	Pointing Device Driver_Version	ZENworks.PointingDeviceDeviceDriver.Version	String	64	Mouse driver version
PointingDevice.Name	Pointing Device_Name	CIM.PointingDevice.Name	String	254	<p>The name of the pointing device, such as Mouse. The string stored in this field will be MOUSE.</p> <p>The CIM.PointingDevice.PointingType field determines the type of the pointing device.</p> <p>The different types of pointing devices are as listed in “Enumeration Values for HARDWARE-Mouse-Name” on page 781.</p>
PointingDevice.Numberofbuttons	Pointing Device_Number of Buttons	CIM.PointingDevice.NumberOfButtons	Unsigned Tiny Integer		The number of buttons used by the pointing device
PointingDevice.IRQNumber	Pointing Device_IRQ Number	CIM.IRQ.IRQNumber	Unsigned Integer		<p>The IRQ channel on the system to which the Mouse pointing device is attached. This information is stored in an IRQ class and not in the PointingDevice class in the database. For more information on how they are associated, see “Understanding the ZENworks for Servers Inventory Database Schema” on page 787.</p>

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
PointingDevice.PointingType	Pointing Device_Type	CIM.PointingDevice.PointingType	Integer (enum)		The pointing device type
ZENKeyboard.Numberoffunction keys	Keyboard_Number of Function Keys	ZENworks.ZENKeyboard.NumberOfFunctionKeys	Unsigned Small Integer		Number of function keys on keyboard
ZENKeyboard.Layout	Keyboard_Layout	ZENworks.ZENKeyboard.Layout	String	254	Layout information. For example, US English.
ZENKeyboard.SubType	Keyboard_Subtype	ZENworks.ZENKeyboard.SubType	Unsigned Integer		A number indicating the subtype of the keyboard
ZENKeyboard.Delay	Keyboard_Delay (mSecs)	ZENworks.ZENKeyboard.Delay	Unsigned Integer		Delay before the repeat of a key
ZENKeyboard.Typeautomaticrate	Keyboard_Typeautomatic Rate (mSecs)	ZENworks.ZENKeyboard.Typeautomatic Rate	Unsigned Integer		Rate of processing the keys
ZENKeyboard.Description	Keyboard_Description	ZENworks.ZENKeyboard.Description	String	254	Keyboard description indicating the type of keyboard. For example, IBM* enhanced (101/102 key) keyboard.
VideoBIOSElement.Manufacturer	Display Driver_Manufacturer	CIM.VideoBIOSElement.Manufacturer	String	254	Manufacturer of the video BIOS driver installed on the system
VideoBIOSElement.Version	Display Driver_Version	CIM.VideoBIOSElement.Version	String	254	Version of the Video BIOS driver
VideoBIOSElement.Install Date	Display Driver_Install Date	CIM.VideoBIOSElement.InstallDate	String	25	Video BIOS release date
VideoBIOSElement.IsShadowed	Display Driver_Is Shadowed	CIM.VideoBIOSElement.ISShadowed	BIT (Used for Boolean conditions)		A Boolean condition indicating if the video BIOS supports shadow memory. 0 represents False and 1 is True.
VideoAdapter.NumberOfcolor planes	Display Adapter_Number of Color Planes	ZENworks.VideoAdapter.NumberOfColorPlanes	Unsigned Integer		Number of color planes supported by the video system
VideoAdapter.CurrentVertical Resolution	Display Adapter_Current Vertical Resolution	ZENworks.VideoAdapter.CurrentVertical Resolution	Unsigned Integer		Vertical resolution of the display

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
VideoAdapter.CurrentHorizontalResolution	Display Adapter_Current Horizontal Resolution	ZENworks.VideoAdpater.Current Horizontal Resolution	Unsigned Integer		Horizontal resolution of the display
VideoAdapter.Description	Display Adapter_Description	ZENworks.VideoAdpater.Description	String	254	Video adapter description
VideoAdapter.MinRefreshRate	Display Adapter_Minimum Refresh Rate	ZENworks.VideoAdpater.MinRefreshRate	Unsigned Integer		Minimum refresh rate of the monitor for redrawing the display, measured in Hertz
VideoAdapter.MaxRefreshRate	Display Adapter_Maximum Refresh Rate	ZENworks.VideoAdpater.MaxRefreshRate	Unsigned Integer		Maximum refresh rate of the monitor for redrawing the display, measured in Hertz
VideoAdapter.VideoArchitecture	Display Adapter_Video Architecture	ZENworks.VideoAdpater.Video Architecture	Unsigned Integer (enum)		The architecture of the video subsystem in this system. For example, CGA/VGA/SVGA/8514A. See “Enumeration Values for HARDWARE-Display Adapter.Video Architecture” on page 781.
VideoAdapter.VideoMemoryType	Display Adapter_Video Memory Type	ZENworks.VideoAdpater.VideoMemory Type	Unsigned Small Integer (Enum)		The type of memory for this adapter. For example, VRAM/SRAM/DRAM/EDORAM. See Enumeration Values for HARDWARE-Display Adapter.Video Memory Type .
VideoAdapter.Maxmemorysupported	Display Adapter_Maximum Memory Supported(KB)	ZENworks.VideoAdpater.MaxMemory Supported	Unsigned Integer		Maximum memory that the display adapter supports for VIDEO RAM
VideoAdapter.CurrentBitsPerPixel	Display Adapter_Current Bits/Pixel	ZENworks.VideoAdpater.CurrentBits PerPixel	Unsigned Integer		Number of adjacent color bits for each pixel
VideoAdapter.ChipSet	Display Adapter_Chip Set	ZENworks.VideoAdpater.ChipSet	String	254	The chip set used in the video adapter

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
VideoAdapter.DACType	Display Adapter_DAC Type	ZENworks.VideoAdpater.DAC Type	String	254	The digital to analog converter type used in the video adapter
VideoAdapter.ProviderName	Display Adapter_Provider	ZENworks.VideoAdapter.Provider	String	254	The manufacturer or the provider name
ZENPOTSModem.Caption	Modem_Caption	ZENworks.ZENPOTSModem.Caption	String	64	The short name of the modem.
ZENPOTSModem.Description	Modem_Description	ZENworks.ZENPOTSModem.Description	String	254	The complete description of the modem. For example, Standard 2400 bps modem, IBM PCMCIA HPC modem.
ZENPOTSModem.Name	Modem_Name	ZENworks.ZENPOTSModem.Name	String	254	The name of the modem dictating its type and usage. For example, Standard Windows Modem means that this is used in standard Windows architecture.
ZENPOTSModem.ProviderName	Modem_Provider	ZENworks.ZENPOTSModem.Provider	String	254	The manufacturer or the provider name
ZENPOTSModem.DeviceID	Modem_Device ID	ZENworks.ZENPOTSModem.DeviceID	String	64	The unique ID assigned to the device
BIOS.BIOSIDBytes	BIOS_BIOS Identification Bytes	ZENworks.BIOS.BIOSIDBytes	String	254	Byte in the BIOS that indicates the computer model
BIOS.SerialNumber	BIOS_Serial Number	ZENworks.BIOS.Serial Number	String	64	Serial number of BIOS assigned by the manufacturer
BIOS.PrimaryBIOS	BIOS_Primary Bios	ZENworks.BIOS.PrimaryBIOS	BIT (Used for Boolean conditions here)		True when set to 1, indicates that this BIOS is the primary BIOS. Used in systems with additional BIOS chips.

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
BIOS.InstallDate	BIOS_Install Date	ZENworks. BIOS.Install Date	String	25	The release date of the BIOS given by the manufacturer
BIOS.Version	BIOS_Version	ZENworks. BIOS.Version	String	254	Version or revision level of the BIOS
BIOS.Manufacturer	BIOS_Manufacturer	ZENworks. BIOS. Manufacturer	String	254	The manufacturer name of BIOS
BIOS.Caption	BIOS_Caption	ZENworks. BIOS.Caption	String	64	The name of the BIOS as given by the BIOS manufacturer
BIOS."size"	BIOS_Size(KB)	ZENworks. BIOS.size	Unsigned Integer		Size of the BIOS in bytes
Processor.CurrentClockSpeed	Processor_Current Clock Speed(MHz)	CIM. Processor. CurrentClockSpeed	Unsigned Integer		Current clock speed of the processor in MHz
Processor.Maxclockspeed	Processor_Maximum Clock Speed(MHz)	CIM. Processor. MaxClock Speed	Unsigned Integer		Maximum clock speed of the processor in MHz
Processor.Role	Processor_Role	CIM. Processor. Role	String	254	Type of processor such as central processor, math coprocessor, and others
Processor.Family	Processor_Processor Family	CIM. Processor. Family	Unsigned Small Integer (enum)		Family the processor belongs to. See “Enumeration Values for HARDWARE-Processor-Family” on page 781.
Processor.Otherfamilydescription	Processor_Other Family Description	CIM. Processor. OtherFamily Description	String	64	Additional description about the processor family, such as the Pentium* processor with MMX technology when the processor cannot be designated using Family.

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
Processor.UpgradeMethod	Processor_Upgrade Method	CIM.Processor.Upgrade Method	Unsigned Small Integer (Enum)		The method by which this processor can be upgraded, if upgrades are supported. See “Enumeration Values for HARDWARE-Processor-Upgrade Method” on page 782.
Processor.Stepping	Processor_Processor Stepping	CIM.Processor.Stepping	String	254	Single-byte code characteristic provided by microprocessor vendors to identify the processor stepping model
Processor.Device ID	Processor_DeviceID	CIM.Processor.DeviceID	String	64	Special hexadecimal string identifying the processor type
CacheMemory.Speed	Cache Memory_Speed(nsec)	CIM.PhysicalMemory.Speed	Unsigned Integer		Speed of this System Cache module in nanoseconds. This is stored in CIM.PhysicalMemory class and is associated to CIM.CacheMemory. For more information on how they are associated, see “Understanding the ZENworks for Servers Inventory Database Schema” on page 787.

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
CacheMemory.Capacity	Cache Memory_Capacity(MB)	CIM.PhysicalMemory.Capacity	Unsigned Integer		Capacity of this System Cache module in nanoseconds. This is stored in CIM.PhysicalMemory class and is associated to CIM.CacheMemory. For more information on how they are associated, see "Understanding the ZENworks for Servers Inventory Database Schema" on page 787.
CacheMemory.Level	Cache Memory_Level	CIM.CacheMemory."Level"	Unsigned Small Integer (enum)		Indicates the cache level: internal cache that is built in to the microprocessors, or external cache that is between the CPU and DRAM.
CacheMemory.WritePolicy	Cache Memory_Write Policy	CIM.CacheMemory.WritePolicy	Unsigned Small Integer (enum)		Indicates the two different ways (Write-Back and Write-Through Cache) that the cache can handle to write to the memory.
CacheMemory.Errormethodology	Cache Memory_Error Methodology	CIM.CacheMemory.Error Methodology	String	254	Error correction scheme supported by this cache component, for example, Parity/ Single Bit ECC/ MultiBit ECC
CacheMemory.Cachetype	Cache Memory_Cache Type	CIM.CacheType	Unsigned Small Integer (enum)		Defines the system cache type. For example, Instruction, Data, Unified.
CacheMemory.LineSize	Cache Memory_Line Size(Bytes)	CIM.CacheMemory.LineSize	Unsigned Integer		Size in bytes of a single cache bucket or line

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
CacheMemory.ReplacementPolicy	Cache Memory_Replacement Policy	CIM.Cache Memory.ReplacementPolicy	Unsigned Integer (enum)		Algorithm that the cache uses to determine which cache lines or buckets should be reused. See “Enumeration Values for HARDWARE-Memory-Cache Memory-Replacement Policy” on page 782.
CacheMemory.ReadPolicy	Cache Memory_Read Policy	CIM.Cache Memory.ReadPolicy	Unsigned Small Integer (enum)		Indicates whether the data cache is for read operation.
CacheMemory.Associativity	Cache Memory_Associativity	CIM.Cache Memory.Associativity	Unsigned Integer (enum)		Defines the system cache associativity (direct-mapped, 2-way, 4-way)
Diskette Drive.Manufacturer	Diskette Drive_Manufacturer	ZENworks.Physical Diskette.Manufacturer	String	254	Vendor name
Diskette Drive.Description	Diskette Drive_Description	ZENworks.Physical Diskette.Description	String	254	Floppy diskette description
Diskette Drive.PhysicalCylinders	Diskette Drive_Physical Cylinders	ZENworks.Physical Diskette.Physical Cylinders	Unsigned Integer		Total number of cylinders or tracks on the floppy
Diskette Drive.PhysicalHeads	Diskette Drive_Physical Heads	ZENworks.Physical Diskette.Physical Heads	Unsigned Small Integer		Number of heads
Diskette Drive.Capacity	Diskette Drive_Capacity (MB)	ZENworks.Physical Diskette.Capacity	Unsigned Integer		Total size

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
Diskette Drive.SectorsPerTrack	Diskette Drive_Sectors/Track	ZENworks.PhysicalDiskette.SectorsPerTrack	Unsigned Integer		Number of sectors per track
Diskette Drive.DeviceID	Diskette Drive_DeviceID	CIM.Diskette Drive	String	64	The drive name representing the floppy drive
ZENDiskDrive.Manufacturer	Physical Disk Drive_Manufacturer	ZENworks.PhysicalDisk.Manufacturer	String	254	Vendor name
ZENDiskDrive.Description	Physical Disk Drive_Description	ZENworks.PhysicalDisk.Description	String	254	Hard disk vendor description
ZENDiskDrive.PhysicalCylinders	Physical Disk Drive_Physical Cylinders	ZENworks.PhysicalDisk.Physical Cylinders	Unsigned Integer		Total number of cylinders
ZENDiskDrive.PhysicalHeads	Physical Disk Drive_Physical Heads	ZENworks.PhysicalDisk.Physical Heads	Unsigned Small Integer		Number of heads
ZENDiskDrive.SectorsPerTrack	Physical Disk Drive_Sectors/Track	ZENworks.PhysicalDisk.SectorsPerTrack	Unsigned Integer		Number of sectors per track
ZENDiskDrive.Capacity	Physical Disk Drive_Capacity(MB)	ZENworks.PhysicalDisk.Capacity	Unsigned Integer		Total size of the hard disk
ZENDiskDrive.Removable	Physical Disk Drive_Removable	ZENworks.LogicalDiskDrive.Removable	BIT		0 indicates that it is a fixed disk and 1 indicates that it is a removable disk.
LocalFileSystem.DeviceID	Logical Disk Drive_Device ID	ZENworks.LogicalDiskDrive.DeviceID	String	64	The drive letter. For example C:, A:, etc.
LocalFileSystem.FileSystemSize	Logical Disk Drive_Size(MB)	CIM.LocalFileSystem.FileSystemSize	Integer		The total size of the file system or the logical disk
LocalFileSystem.AvailableSpace	Logical Disk Drive_Free Size(MB)	CIM.LocalFileSystem.AvailableSpace	Integer		The available size of the file system or the logical disk
LocalFileSystem.VolumeSerial Number	Logical Disk Drive_Volume Serial Number	CIM.LocalFileSystem.VolumeSerialNumber	String	254	The volume serial number of the specified drive.
LocalFileSystem.Caption	Logical Disk Drive_Caption	CIM.LocalFileSystem.Caption	String	64	The volume label of the specified drive

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
LocalFileSystem.FileSystemType	Logical Disk Drive_ File System Type	CIM.LocalFileSystem.FileSystemType	String	254	The file system on the drive. For example, FAT, NTFS, etc.
CDROMDrive.Manufacturer	CDROM_Manufacturer	ZENworks. Physical CDROM. Manufacturer	String	254	The manufacturer of the CD-ROM drive
CDROMDrive.Caption	CDROM_Caption	ZENworks. Physical CDROM. Caption	String	64	CD-ROM label
CDROMDrive.Description	CDROM_ Description	ZENworks. Physical CDROM. Description	String	254	Description of the CD-ROM drive, as given by the manufacturer. For example, ATAPI CDROM, CREATIVE CD1620E SL970520.
CDROMDrive.DeviceID	CDROM_ Device ID	ZENworks. Logical CDROM. DeviceID	String	64	Drive letter allocated for the CD-ROM on the inventoried server
SerialPort.Name	Serial Port_Name	ZENworks. SerialPort. Name	String	254	The name of the serial port. For example, COM1, COM2 and others.
SerialPort.Address	Serial Port_ Address	ZENworks. SerialPort. Address	Unsigned Integer		The address mapped in memory for the serial port

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
SerialPort.IRQNumber	Serial Port_IRQ Number	CIM.IRQ.IRQNumber	Unsigned Integer		<p>The IRQ channel on the system to which the serial port is attached. In the database, this information is stored in an IRQ class and not in Serial Port class.</p> <p>For more information on how they are associated, see Chapter 28, "Understanding the ZENworks for Servers Inventory Database Schema," on page 787.</p>
ParallelPort.Name	Parallel Port_Name	ZENworks. ParallelPort. Name	String	254	The name of the parallel port. For example, LPT1 and others.
ParallelPort.Address	Parallel Port_ Address	ZENworks. ParallelPort. Address	Unsigned Integer		The name of the parallel port. For example, LPT1 and others
ParallelPort.DMASupport	Parallel Port_DMA Support	ZENworks. ParallelPort. DMASupport	BIT (used for Boolean condition s here)		If True or 1, then it means that DMA is channel is allocated for bulk data transfer for use with devices connected to the parallel ports

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
ParallelPort.IRQNumber	Parallel Port_IRQ Number	CIM.IRQ.IRQNumber	Unsigned Integer		The IRQ channel on the system to which the parallel port is attached. This information is stored in an IRQ class and not in a parallel port class in the database. For more information on how they are associated, see Chapter 28, "Understanding the ZENworks for Servers Inventory Database Schema," on page 787.
Bus.Version	Bus_Version	ZENworks.Bus.BusVersion	String	254	Version of the bus supported by the inventoried server
Bus.Description	Bus_Description	ZENworks.Bus.Description	String	254	Description of the bus.
Bus.BusType	Bus_Bus Type	ZENworks.Bus.BusType	Integer (enum)		The bus type of the system
Bus.Name	Bus_Name	ZENworks.Bus.Name	String	254	Name of the internal system bus
Bus.DeviceID	Bus_Device ID	ZENworks.Bus.DeviceID	String	64	The unique ID for the specific bus
ZENNetworkAdapter.Name	Network Adapter_Name	CIM.ZENworks.ZENAdapter.Name	String	254	Network adapters installed on the system
ZENNetworkAdapter.MaxSpeed	Network Adapter_Max_Speed (Mbps)	CIM.ZENworks.ZENAdapter.MaxSpeed	Unsigned Integer		Rate at which the adapter can transfer data
ZENNetworkAdapter.PermanentAddress	Network Adapter_Permanent Address	CIM.ZENworks.ZENAdapter.PermanentAddress	String	64	Machine address stored permanently in the adapter (MAC address)
ZENNetworkAdapter.MACAddress	Network Adapter_Address	CIM.ZENworks.ZENAdapter.MACAddress	String	64	The MAC address stored in the network adapter
ZENNetworkAdapter.ProviderName	Network Adapter_Provider	CIM.ZENworks.ZENAdapter.Provider	String	254	The manufacturer or the provider

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
ZENNetworkAdapter.Adapter Type	Network Adapter_ Adapter Type	CIM.ZENworks.ZENAdapte r. AdapterType	String	254	Type of the adapter such as Ethernet or FDDI adapter
SoundAdapter.Description	Multimedia Card_ Description	ZENworks. SoundAdapter. Description	String	254	Description of the multimedia component for the inventoried server
SoundAdapter.Name	Multimedia Card_ Name	ZENworks. SoundAdpater. Name	String	254	Name of the sound card installed on the system
SoundAdapter.Manufacturer	Multimedia Card_ Manufacturer	ZENworks. SoundAdapter. Manufacturer	String	254	Vendor name
SoundAdapter.ProviderName	Multimedia Card_ Provider	ZENworks. SoundAdapter. Provider	String	254	The provider or the manufacturer of the multimedia card
Battery.Name	Battery_ Name	CIM.Battery. Name	String	254	Name of the battery installed on the system
Battery.Chemistry	Battery_ Chemistry	CIM.Battery. Chemistry	Unsigned Small Integer		Indicates battery's chemistry, such as lead acid, nickel cadmium and others. See "Enumeration Values for HARDWARE-Battery-Chemistry" on page 781.
Battery.DesignCapacity	Battery_ Design Capacity(mWatt-hours)	CIM.Battery. Design Capacity	Unsigned Integer		The design capacity of the battery in mWatt-hours
Battery.DesignVoltage	Battery_ Design Voltage(MilliVolts)	CIM.Battery. DesignVoltage	Unsigned Integer		The design voltage of the battery in mVolts
Battery.SmartBatteryVersion	Battery_ Smart Battery Version	CIM.Battery. SmartBatteryVersion	String	64	The Smart Battery Data Specification version number supported by this battery
Battery.Manufacturer	Battery_ Manufacturer	CIM.PhysicalComponent. Manufacturer	String	254	Vendor name of the battery
Battery.InstallDate	Battery_ Install Date	CIM.PhysicalComponent. InstallDate	String	25	Date of manufacturing the battery

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
Battery.SerialNumber	Battery_Serial Number	CIM.PhysicalComponent.SerialNumber	String	64	Battery serial number
PowerSupply.Description	Power Supply_Description	CIM.PowerSupply.Description	String	254	Name and description of the power supply on the system
PowerSupply.TotalOutputPower	Power Supply_Total Output Power (MilliWatts)	CIM.PowerSupply.TotalOutputPower	Unsigned Integer		Total output power of the power supply
IPProtocolEndPoint.Address	IP Address_Address	CIM.IPProtocolEndpoint.Address	String	254	IP address of the inventoried server
IPProtocolEndPoint.SubnetMask	IP Address_Subnet Mask	CIM.IPProtocolEndpoint.SubnetMask	String	254	The subnet mask of the inventoried server
DNSName.LABEL	DNS_LABEL	ManageWise.DNSName.Label	String	254	DNS name of the inventoried server
IPXProtocolEndPoint.Addresses	IPX Address_Address	CIM.IPXProtocolEndpoint.Address	String	254	IPX address of the inventoried server
LANEndPoint.MACAddress	MAC Address_Address	CIM.LANEndpoint.MACAddress	String	12	MAC address of the inventoried server
MotherBoard.Version	MotherBoard_Version	ZENworks.Motherboard.Version	String	64	Motherboard version
MotherBoard.Description	MotherBoard_Description	ZENworks.Motherboard.Description	String	254	The description of the motherboard
MotherBoard.Manufacturer	MotherBoard_Manufacturer	ZENworks.Motherboard.Manufacturer	String	254	The manufacturer of the motherboard
MotherBoard.NumberOfSlots	MotherBoard_Number Of Slots	ZENworks.Motherboard.Numberofslots	Integer		The number of expansion slots on the motherboard
IRQ.Number	IRQ_IRQ Number	CIM.IRQ.IRQNumber	Unsigned Integer		The system interrupt number

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
IRQ.Availability	IRQ_Availability	CIM.IRQ. Availability	Unsigned Small Integer (Enum)		Indicates whether the IRQ channel is used or available. Enumeration values are as follows: 1 = "Other" 2 = "Unknown" 3 = "Available" 4 = "In Use/Not Available" 5 = "In Use and Available/ Shareable"
IRQ.TriggerType	IRQ_IRQ Trigger Type	CIM.IRQ. TriggerType	Unsigned Small Integer		IRQ trigger type indicating whether edge (value=4) or level triggered (value=3) interrupts occur. Enumeration values are as follows: 1 = "Other" 2 = "Unknown" 3 = "Level" 4 = "Edge"
IRQ.Shareable	IRQ_IRQ Shareable	CIM.IRQ. Shareable	Unsigned Small Integer		Boolean indicating whether the IRQ can be shared
SLOT.MaxDataWidth	Slot_Maximum Data Width	CIM.Slot. MaxData Width	Unsigned Small Integer		Maximum bus width of adapter cards that can be inserted into this slot in bits. If the value is 'unknown', enter 0. If the value is other than 8, 16, 32, 64 or 128, enter 1. It is expressed in bits
SLOT.ThermalRating	Slot_Thermal Rating (MilliWatts)	CIM.Slot. Thermal Rating	Unsigned Integer		Maximum thermal dissipation of the slot in milliwatts
SLOT.Description	Slot_Description	CIM.SlotDescription	String	254	The description of the adapter mounted on the slot
DMA.DMAChannel	DMA_DMA Channel Number	CIM.DMA. DMAChannel	Unsigned Integer		The DMA channel number

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
DMA.Description	DMA_Description	CIM.DMA.Description	String	254	The name of the device using the DMA channel
DMA.Availability	DMA_Availability	CIM.DMA.Availability	Unsigned Small Integer		Indicates whether the DMA channel is available or not: Enumeration values are as follows: 1 = "Other" 2 = "Unknown" 3 = "Available" 4 = "In Use/Not Available" 5 = "In Use and Available/Shareable"
DMA.BurstMode	DMA_DMA Burst Mode	CIM.DMA.BurstMode	BIT (used for Boolean condition here)		Indication that the DMA channel supports the burst mode
NetWareOperatingSystem.Version	NetWare.Version	ZENworks.NetWareOperatingSystem.Version	String	254	Version of the NetWare operating system
NetWareOperatingSystem.CodePage	NetWare.Code Page	ZENworks.NetWareOperatingSystem.CodePage	String	254	The operating system language code page setting.
NetWareOperatingSystem.Caption	NetWare.Caption	ZENworks.NetWareOperatingSystem.Caption	String	64	Short name for the NetWare server. For example, NetWare
NetWareOperatingSystem.InstallDate	NetWare.Install Date	ZENworks.NetWareOperatingSystem.InstallDate	String	25	The install date of the NetWare
NetWareOperatingSystem.AccountingVersion	NetWare.Accounting Version	ZENworks.NetWareOperatingSystem.AccountingVersion	String	254	The version of the NetWare Accounting system
NetWareOperatingSystem.InternetBridgeSupport	NetWare.Internet Bridge Support	ZENworks.NetWareOperatingSystem.MaximumNumberOfConnections	String	254	Maximum number of connections allowed
NetWareOperatingSystem.MaximumNumberOfConnections	NetWare.Maximum Number Of Connections	ZENworks.NetWareOperatingSystem.MaximumNumberOfConnections	Integer		Maximum number of connections allowed
NetWareOperatingSystem.MaximumNumberOfVolumes	NetWare.Maximum Number Of Volumes	ZENworks.NetWareOperatingSystem.MaximumNumberOfVolumes	Integer		Maximum number of volumes allowed

Export Wizard Attribute Name	Export Attribute Name (Column Heading in the .CSV file)	Database Schema Attribute Name	Data Type	Length	Description of the Attribute
NetWareOperatingSystem.PeakConnectionsUsed	NetWare.Peak Connections Used	ZENworks.NetWareOperatingSystem.PeakConnectionsUsed	Integer		Maximum number of connections used
NetWareOperatingSystem.PrintServerVersion	NetWare.Print Server Version	ZENworks.NetWareOperatingSystem.PrintServerVersion	String	254	The print server version if the server is used as a print server
NetWareOperatingSystem.QueueingVersion	NetWare.Queueing Version	NetWareOperatingSystem.QueueingVersion	String	254	The NetWare Print server's print queue version if this server is used as a print server
NetWareOperatingSystem.RevisionLevel	NetWare.Revision Level	ZENworks.NetWareOperatingSystem.RevisionLevel	String	254	The revision level of NetWare
NetWareOperatingSystem.SecurityRestrictionLevel	NetWare.Security Restriction Level	ZENworks.NetWareOperatingSystem.SecurityRestrictionLevel	String	254	The security restriction level
NetWareOperatingSystem.SFTLevel	NetWare.SFT Level	ZENworks.NetWareOperatingSystem.SFTLevel	String	254	The SFT level
NetWareOperatingSystem.TTSLevel	NetWare.TTS Level	ZENworks.NetWareOperatingSystem.TTSLevel	String	254	The TTS level
NetWareOperatingSystem.VAPVersion	NetWare.VAP Version	ZENworks.NetWareOperatingSystem.VAPVersion	String	254	The VAP version
NetWareOperatingSystem.VirtualConsoleVersion	NetWare.Virtual Console Version	ZENworks.NetWareOperatingSystem.VirtualConsoleVersion	String	254	The virtual console version
NetWareOperatingSystem.InternalNetworkNumber	NetWare.Internal Network Number	ZENworks.NetWareOperatingSystem.InternalNetworkNumber	String	254	The internal network number reported by Netware
NetWareOperatingSystem.TotalVisibleMemorySize	NetWare.Total Memory(MB)	ZENworks.NetWareOperatingSystem.TotalVisibleMemorySize	String	254	The total primary memory as reported by NetWare
NetWareOperatingSystem.TotalVirtualMemorySize	NetWare.Total Virtual Memory(MB)	ZENworks.NetWareOperatingSystem.TotalVirtualMemorySize	String	254	The total virtual memory size used by NetWare

Enumeration Values for HARDWARE-Display Adapter.Video Architecture

The enumeration values are:

1 = "Other"	6 = "SVGA"	11 = "XGA"
2 = "Unknown"	7 = "MDA"	12 = "Linear Frame Buffer"
3 = "CGA"	8 = "HGC"	160 = "PC-98"
4 = "EGA"	9 = "MCGA"	
5 = "VGA"	10 = "8514A"	

Enumeration Values for HARDWARE-Display Adapter.Video Memory Type

The enumeration values are:

1 = "Other"	6 = "WRAM"	11 = "3DRAM"
2 = "Unknown"	7 = "EDO RAM"	12 = "SDRAM"
3 = "VRAM"	8 = "Burst Synchronous DRAM"	13 = "SGRAM"
4 = "DRAM"	9 = "Pipelined Burst SRAM"	
5 = "SRAM"	10 = "CDRAM"	

Enumeration Values for HARDWARE-Mouse-Name

The enumeration values are:

1 = "Other"	4 = "Track Ball"	7 = "Touch Pad"
2 = "Unknown"	5 = "Track Point"	8 = "Touch Screen"
3 = "Mouse"	6 = "Glide Point"	9 = "Mouse - Optical Sensor"

Enumeration Values for HARDWARE-Battery-Chemistry

The enumeration values are:

1 = "Other"	5 = "Nickel Metal Hydride"
2 = "Unknown"	6 = "Lithium-ion"
3 = "Lead Acid"	7 = "Zinc air"
4 = "Nickel Cadmium"	8 = "Lithium Polymer"

Enumeration Values for HARDWARE-Processor-Processor Family

The enumeration values are:

1 = "Other"	15 = "Celeron(TM)"	130 = "Itanium(TM) Processor"
2 = "Unknown"	16 = "Pentium(R) II Xeon(TM)"	176 = "Pentium(R) III Xeon(TM)"
11 = "Pentium(R) Brand"	17 = "Pentium(R) III"	177 = "Pentium(R) III Processor with Intel(R) SpeedStep(TM) Technology"

12 = "Pentium(R) Pro"	24 = "AMD Duron(TM) Processor Family"	178 = "Pentium(R) 4 Processor"
13 = "Pentium(R) II"	29 = "AMD Athlon(TM) Processor Family"	
14 = "Pentium(R) Processor with MMX(TM) Technology"	30 = "AMD29000 Family"	

Enumeration Values for **HARDWARE-Processor-Upgrade Method**

The enumeration values are:

1 = "Other"	5 = "Replacement/Piggy Back"	9 = "Slot 2"
2 = "Unknown"	6 = "None"	10 = "370 Pin Socket"
3 = "Daughter Board"	7 = "LIF Socket"	11 = "Slot A"
4 = "ZIF Socket"	8 = "Slot 1"	12 = "Slot M"

Enumeration Values for **HARDWARE-Memory-Cache Memory-Replacement Policy**

The enumeration values are:

1 = "Other"	5 = "Last In First Out (LIFO)"
2 = "Unknown"	6 = "Least Frequently Used (LFU)"
3 = "Least Recently Used (LRU)"	7 = "Most Frequently Used (MFU)"
4 = "First In First Out (FIFO)"	8 = "Data Dependent Multiple Algorithm"

Enumeration Values for **SOFTWARE-Operating Systems-Name**

The enumeration values are:

0 = "Unknown"	17 = "WIN98"	58 = "Windows 2000"
1 = "Other"	18 = "WINNT"	59 = "Dedicatedo"
16 = "WIN95"	21 = "NetWare"	63 = "Windows (R) Me"

Enumeration Values for **HARDWARE-Bus-Protocol Supported**

The enumeration values are:

0 = "Internal"	6 = "VME Bus"	12 = "Internal Processor"
1 = "ISA"	7 = "NuBus"	13 = "Internal Power Bus"
2 = "EISA"	8 = "PCMCIA Bus"	14 = "PNP ISA Bus"
3 = "MicroChannel"	9 = "C Bus"	15 = "PNP Bus"
4 = "TurboChannel"	10 = "MPI Bus"	16 = "Maximum Interface Type"
5 = "PCI Bus"	11 = "MPSA Bus"	

Enumeration Values for GENERAL-Asset-Management Technology

The enumeration values are:

1 = "Unknown"	3 = "DMI Enabled"	5 = "SNMP Enabled"
2 = "Other"	4 = "WMI Enabled"	6 = "DMI and WMI Enabled"

Enumeration Values for SOFTWARE-Operating Systems-Windows-Role

The enumeration values are:

0 = "Unknown"	2 = "Managed Servers"
1 = "Other"	3 = "Managed Workstation"

Enumeration Values for SOFTWARE-Scanner Information-Scan Mode

The enumeration values are:

1 = "Unknown"	3 = "DMI "	5 = "SNMP"
2 = "Other"	4 = "WMI "	6 = "DMI & WMI "

Enumeration Values for HARDWARE-Processor-Role

The enumeration values are:

1 = "Other"	3 = "Central Processor "	5 = "DSP Processor"
2 = "Unknown"	4 = "Math Processor "	6 = "Video Processor "

Enumeration Values for HARDWARE-Processor-Upgrade Method

The enumeration values are:

1 = "Other"	5 = "Replacement/Piggy Back "	9 = "Slot 2"
2 = "Unknown"	6 = "None "	10 = "370 Pin Socket"
3 = "Daughter Board"	7 = "LIF Socket"	11 = "Slot A"
4 = "ZIF Socket "	8 = "Slot 1"	12 = "Slot M"

Enumeration Values for SYSTEM-Cache Memory-Level

The enumeration values are:

1 = "Other"	3 = "Primary "	5 = "Tertiary"
2 = "Unknown"	4 = "Secondary "	6 = "Not Applicable"

Enumeration Values for SYSTEM-Cache Memory-Level

The enumeration values are:

1 = "Other"	3 = "Write Back "	5 = "Varies with Address"
2 = "Unknown"	4 = "Write Through "	6 = "Determination Per I/O"

Enumeration Values for SYSTEM-Cache Memory-Cache Type

The enumeration values are:

1 = "Other"	3 = "Instruction "	5 = "Unified"
2 = "Unknown"	4 = "Data "	

Enumeration Values for SYSTEM-Cache Memory-Replacement Policy

The enumeration values are:

1 = "Other"	4 = "First In First Out (FIFO) "	7 = "Most Frequently Used (MFU)"
2 = "Unknown"	5 = "Last In First Out (LIFO) "	8="Data Dependent Multiple Algorithms"
3 = "Least Recently Used (LRU)"	6 = "Least Frequently Used (LFU) "	

Enumeration Values for SYSTEM-Cache Memory-Read Policy

The enumeration values are:

1 = "Other"	3 = "Read "	5 = "Read and Read-ahead"
2 = "Unknown"	4 = "Read-ahead "	6="Determination Per I/O"

Enumeration Values for SYSTEM-Cache Memory-Associativity

The enumeration values are:

1 = "Other"	4 = "2-way Set-Associative "	7 = "8-way Set-Associative"
2 = "Unknown"	5 = "4-way Set-Associative"	8="16-way Set-Associative"
3 = "Direct Mapped"	6 = "Fully Associative"	

Enumeration Values for SYSTEM-IRQ-Availability

The enumeration values are:

1 = "Other"	3 = "Available "	5 = "In Use and Available/Shareable"
2 = "Unknown"	4 = "In Use/Not Available "	

Enumeration Values for SYSTEM-IRQ-IRQ Trigger Type

The enumeration values are:

1 = "Other"

3 = "Level "

2 = "Unknown"

4 = "Edge "

Enumeration Values for SYSTEM-DMA-Availability

The enumeration values are:

1 = "Other"

3 = "Available "

5 = "In Use and Available/
Shareable"

2 = "Unknown"

4 = "In Use/Not Available "

28

Understanding the ZENworks for Servers Inventory Database Schema

This section describes the design of the Novell® ZENworks® for Servers (ZfS) Inventory database schema implemented using the Common Information Model (CIM) of Distributed Management Task Force (DMTF). To understand this section effectively, you should be familiar with terminology such as CIM and Desktop Management Interface (DMI). You should also have a solid understanding of Relational Database Based Managed Systems (RDBMS) and database concepts.

The following sections provide in-depth information:

- ♦ “Overview” on page 787
- ♦ “CIM Schema” on page 788
- ♦ “Inventory Database Schema in ZfS” on page 798

Overview

The DMTF is the industry organization leading the development, adoption, and unification of management standards and initiatives for desktop, enterprise, and Internet environments. For more information about DMTF, see the [DMTF Web site \(http://www.dmtf.org\)](http://www.dmtf.org).

The DMTF CIM is an approach to system and network management that applies the basic structuring and conceptualization techniques of the object-oriented paradigm. The approach uses a uniform modeling formalism that together with the basic repertoire of object-oriented constructs supports the cooperative development of an object-oriented schema across multiple organizations.

A management schema is provided to establish a common conceptual framework at the level of a fundamental topology, both with respect to classification and association, and to a basic set of classes intended to establish a common framework for a description of the managed environment. The management schema is divided into the following conceptual layers:

- ♦ **Core Model:** An information model that captures notions that are applicable to all areas of management.
- ♦ **Common Model:** An information model that captures notions that are common to particular management areas, but independent of a particular technology or implementation. The common areas are systems, applications, databases, networks, and devices. The information model is specific enough to provide a basis for the development of management applications. This model provides a set of base classes for extension into the area of technology-specific schema. The Core and Common models together are expressed as the CIM schema.
- ♦ **Extension Schema:** This schema represents technology-specific extensions of the Common model. These schema are specific to environments, such as operating systems, for example, NetWare®, UNIX*, or Microsoft* Windows*.

CIM comprises a specification and a schema (see the [DMTF Web site \(http://www.dmtf.org/standards/standard_cim.php\)](http://www.dmtf.org/standards/standard_cim.php)). The specification defines the meta-schema plus a concrete representation language called Managed Object Format (MOF).

CIM Schema

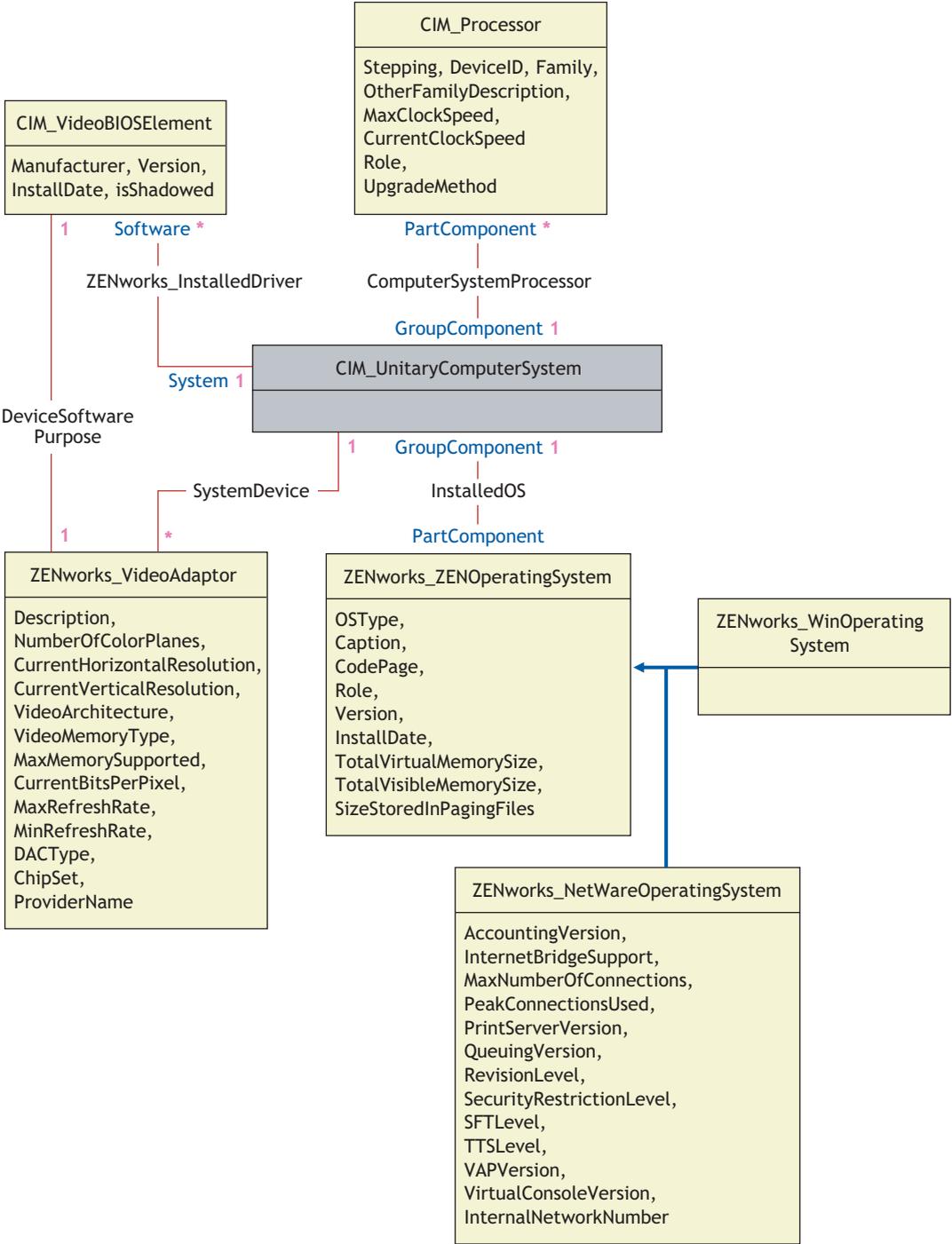
The elements of the meta schema are classes, properties, and methods. The meta schema also supports indications and associations as types of classes and references as types of properties.

Classes can be arranged in a generalization hierarchy that represents subtype relationships between classes. The generalization hierarchy is a rooted, directed graph that does not support multiple inheritance.

A regular class may contain scalar or array properties of any intrinsic type such as Boolean, integer, string, and others. It cannot contain embedded classes or references to other classes.

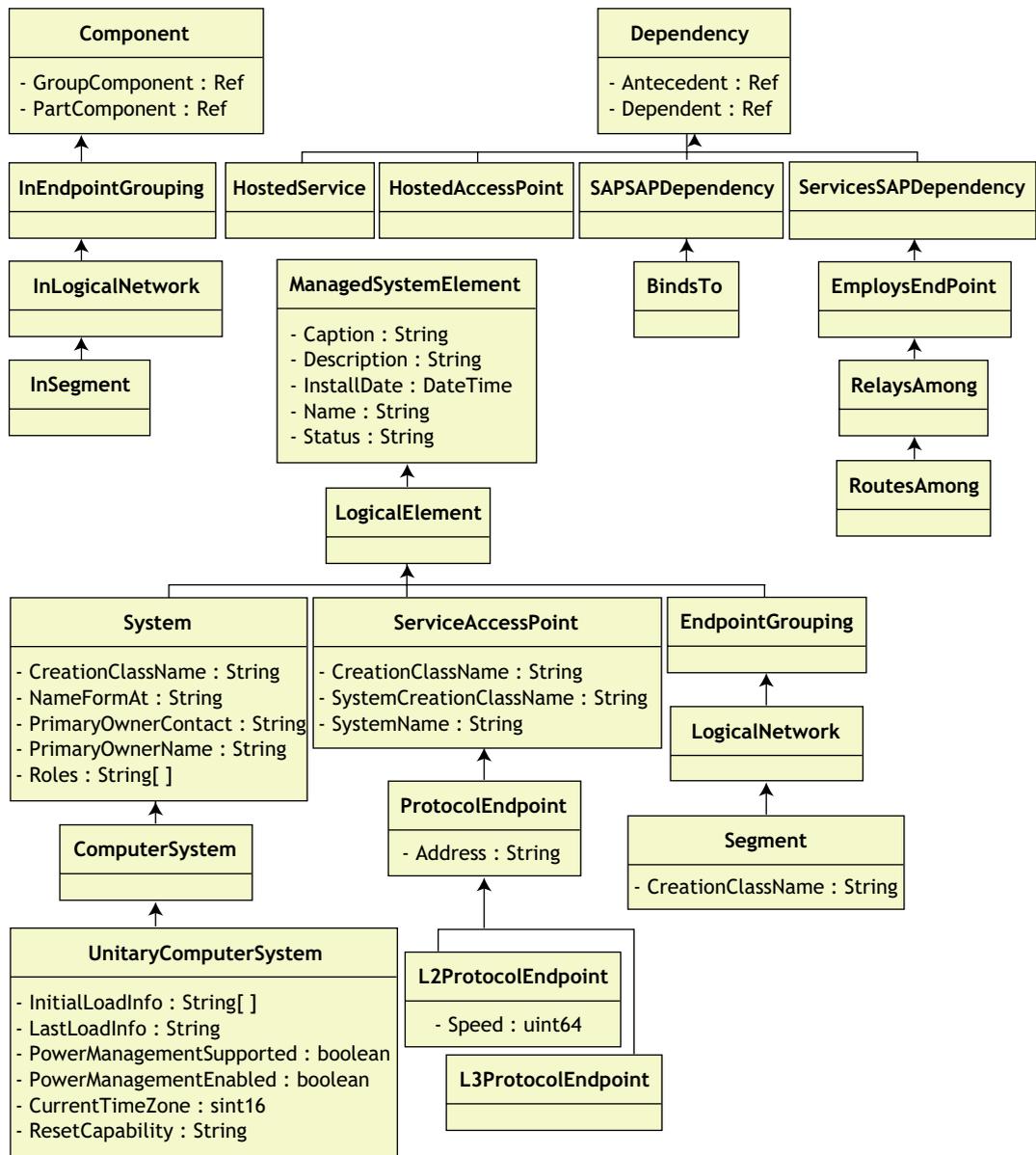
An association is a special class that contains two or more references. It represents a relationship between two or more objects. Because of the way associations are defined, it is possible to establish a relationship between classes without affecting any of the related classes. That is, addition of an association does not affect the interface of the related classes. Only associations can have references.

The schema fragment in the following illustration shows the relationships between some CIM objects that ZfS uses.



The illustration shows how the CIM schema maps to a relational DBMS schema. The classes are shown with the class name as the box heading. The associations are labeled within the lines between two classes.

The inheritance hierarchy of this schema fragment is shown in the following illustration of the CIM 2.2 schema. The references shown as type Ref are in bold with each association sub-type narrowing the type of the reference.



CIM-to-Relational Mapping

CIM is an object model complete with classes, inheritance, and polymorphism. The generated mapping to a relational schema preserves these features to the maximum extent. The following two aspects are part of the relational mapping:

- ◆ **Logical Schema:** The logical schema defines how the data appears to applications, similar to an API. The goal is that the logical schema remains the same irrespective of the underlying database so that application software can run unchanged on any supported databases. Though SQL (pronounced as sequel) is a standard, this goal is not fully possible. Application software will need to know more about the database in use and this information can be abstracted and isolated to a small area of the application code.
- ◆ **Physical Schema:** The physical schema defines how the data is structured in the database. The schema tends to be specific to the database because of the nature of SQL and RDBMS. This document will describe the physical schema in general terms only.

A table in the database represents each class in the CIM hierarchy. A column of the appropriate type in the table represents each non-inherited property in the class. Each table also has a primary key, id\$, which is a 64-bit integer that uniquely identifies an instance. An instance of a CIM class is represented by a row in each table that corresponds to a class in its inheritance hierarchy. Each row has the same value for id\$.

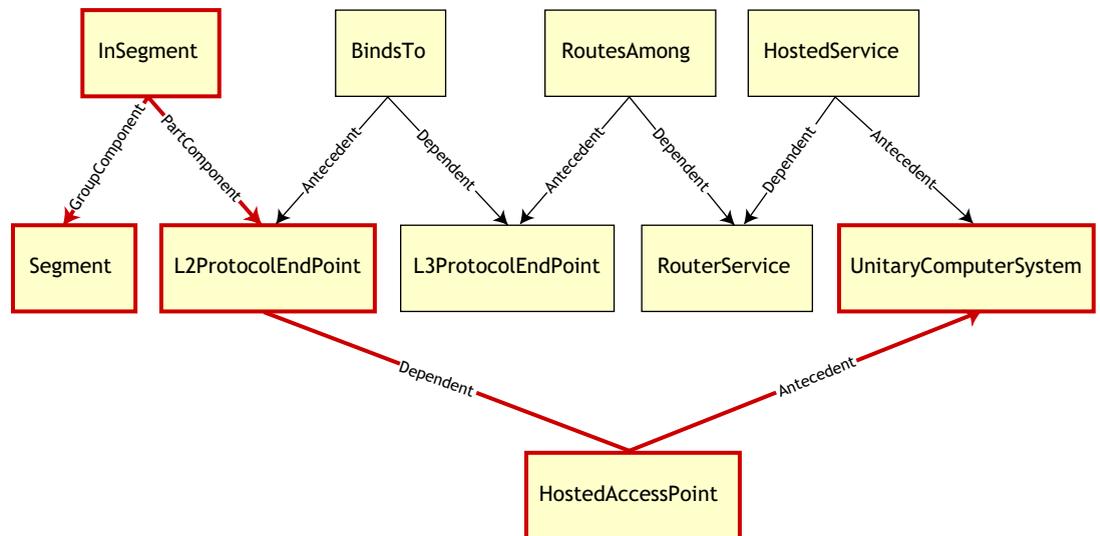
Each CIM class is also represented by a view that uses id\$ to join rows from the various tables in the inheritance hierarchy to yield a composite set of properties (inherited plus local) for an instance of that class. The view also contains an extra column, class\$, of type integer that represents the type of the actual (leaf-most) class of the instance.

Associations are mapped in the same manner as regular classes, with a reference property being represented by a column with the id\$ field of the referenced object instance. Thus, associations can be traversed by doing a join between the reference field in the association and the id\$ field in the referenced table.

The following illustration depicts a typical query using this mapping:

Get Computers for Segment

```
SELECT CIM.UnitaryComputerSystem.*
FROM   CIM.UnitaryComputerSystem, CIM.Segment, CIM.L2ProtocolEndPoint,
       CIM.HostedAccessPoint, CIM.InSegment
WHERE  CIM.SegmentName = 'xxx'
AND    CIM.InSegment.GroupComponent = CIM.Segment.id$
AND    CIM.InSegment.PartComponent = CIM.L2ProtocolEndPoint.id$
AND    CIM.HostedAccessPoint.Dependent = CIM.L2ProtocolEndPoint.id$
AND    CIM.HostedAccessPoint.Antecedent = CIM.UnitaryComputerSystem.id$
```



This query finds all the computers attached to a given network segment. The classes and relationships involved are highlighted with borders.

The following topics describe both the schema types:

- ◆ “Logical Schema” on page 792
- ◆ “Physical Schema” on page 798

Logical Schema

The logical schema is the database schema as seen by users of the database and the application program. The schema consists of stored procedures and views. The underlying tables are not visible to the application.

Typically, each CIM class has the following:

- ◆ A constructor procedure to generate an instance of the class. For more information, see [“Constructor” on page 796](#).
- ◆ A destructor procedure to destroy an instance of the class. For more information, see [“Destructor” on page 798](#).
- ◆ A view to access and update the values of properties of the class.

ZfS Inventory components use JDBC to issue SQL statements to the RDBMS and to convert between RDBMS data types and Java* data types. The use of JDBC with stored procedures and views provides a level of abstraction that insulates application code from the underlying database technology and from changes to the physical schema.

The various elements of the logical schema are discussed in more detail in the following sections:

- ◆ [“Naming Schema Elements” on page 792](#)
- ◆ [“Users and Roles” on page 793](#)
- ◆ [“Data Types” on page 793](#)
- ◆ [“Views” on page 794](#)
- ◆ [“Object Identifier Id\\$” on page 795](#)
- ◆ [“Constructor” on page 796](#)
- ◆ [“Destructor” on page 798](#)

Naming Schema Elements

We recommend that you use the CIM names unchanged in the database schema. Some problems may still ensue because of the differences in the naming schemes, such as the following:

- ◆ Names in CIM and SQL are not case sensitive.
- ◆ All databases have different sets of reserved words that must be enclosed in quotes (" ") when used as schema element names; however, in Oracle*, enclosing a name in quotes makes it case sensitive.
- ◆ CIM classes avoid using SQL reserved words as names.
- ◆ CIM names are not limited in length and usually the names are long. Sybase allows up to 128 characters, but Oracle restricts the names to 30 characters.

Most of these problems are avoided during schema generation by preserving the case of CIM names, abbreviating any names longer than 30 characters, and placing quotes around any name that is in the union of the sets of reserved words.

Any name longer than 28 characters is abbreviated to a root name of 28 or fewer characters to allow a two-character prefix so that all associated SQL schema elements can use the same root name. The abbreviation algorithm shortens a name so that it is mnemonic, recognizable, and also unique within its scope. The abbreviated name is given a # character as a suffix (note that # is an illegal character in CIM) to prevent clashes with other names. If two or more names within the

same scope generate the same abbreviation, an additional digit is appended to make the name unique. For example, AttributeCachingForRegularFilesMin is abbreviated to AttCacForRegularFilesMin#.

All such mangled names are written to the mangled name table so that a program can look up the real CIM name and retrieve the mangled name to use with the SQL.

Views are the schema elements that are most often manipulated by application code and queries. They use the same name as the CIM class they represent. For example, the CIM_UnitaryComputerSystem class is represented by a view named CIM.UnitaryComputerSystem.

When necessary, names for indexes and auxiliary tables are created by concatenating the class name and property name separated by a \$ character. These names are usually abbreviated. For example, NetworkAdapter\$NetworkAddresses is abbreviated to NetAdapter\$NetAddresses#. This does not have any adverse impact on ZfS schema users.

Users and Roles

In SQL, a user with the same name as the schema is the owner of each schema, for example, CIM, ManageWise[®], ZENworks[®], and others.

Additionally, there is an MW_DBA user that has Database Administrator privileges and rights to all schema objects. The MW_Reader role has read-only access to all schema objects and the MW_Updater role has read-write-execute access to all schema objects.

Application programs should access the database as either MW_Reader or MW_Updater for a Sybase database and MWO_Reader or MWO_Updater for an Oracle database, depending on their requirements.

Data Types

CIM data types are mapped to the most appropriate data type provided by the database. Usually, the Java application does not require the type because it uses JDBC to access the data.

Java does not natively support unsigned types, so you should use classes or integer types of the next size to represent them. Also, ensure that there are no problems while reading or writing to the database. For example, reading or writing a negative number to an unsigned field in the database is likely cause an error.

Strings in CIM and Java are Unicode*, so the database is created using the UTF8 character set. Internationalization does not pose any problems; however, it may create problem with case sensitivity in queries.

All databases preserve the case of string data stored within them, but may access the data as either case sensitive or otherwise during queries. In ZfS, the Inventory Query component is not affected because the queried data is retrieved from the database before being queried and so case sensitivity is automatically taken care of.

In CIM, strings may be specified with or without a maximum size in characters. Many strings have no specified size, which means they can be unlimited in size. For efficiency reasons, these unlimited strings are mapped to a variable string with maximum size of 254 characters. CIM strings with a maximum size are mapped to variable database strings of the same size. The size in the database is in bytes and not as characters because a Unicode character may require more than one byte for storage.

Views

Each CIM class is represented in the database by a view that contains all the local and inherited non-array properties of that class. The view is named the same as the CIM class. For example, the CIM class CIM_System represents a SQL view named CIM.System, as shown in the following illustration.

The CIM.System view is created with attributes that are selected from multiple tables. These attributes include: id\$ selected from cim.t\$ManagedSystemElement, class\$ is filled up automatically using the function mw_dba.extractClass, Caption selected from cim.t\$ManagedSystemElement, Description selected from cim.t\$ManagedSystemElement, InstallDate selected from cim.t\$ManagedSystemElement, Status selected from cim.t\$ManagedSystemElement, CreationClassName selected from cim.t\$System, Name selected from cim.t\$ManagedSystemElement. NameFormat selected from cim.t\$System.NameFormat, PrimaryOwnerContact selected from cim.t\$System, and PrimaryOwnerName selected from cim.t\$System. The view is created by joining the tables CIM.t\$ManagedSystemElement and CIM.t\$System where the id\$ of both the tables are same.

The CIM.SYSTEM view is as follows:

```
CREATE VIEW CIM.System
{
    id$,
    class$,
    Caption,
    Description,
    InstallDate,
    Status,
    CreationClassName,
    Name,
    NameFormat,
    PrimaryOwnerContact,
    PrimaryOwnerName
}
AS SELECT
    CIM.t$ManagedSystemElement.id$,
    MW_DBA.extractClass(CIM.t$ManagedSystemElement.id$),
    CIM.t$ManagedSystemElement.Caption,
    CIM.t$ManagedSystemElement.Description,
    CIM.t$ManagedSystemElement.InstallDate,
    CIM.t$ManagedSystemElement.Status,
    CIM.t$System.CreationClassName,
    CIM.t$ManagedSystemElement.Name,
```

```

CIM.t$System.NameFormat ,
CIM.t$System.PrimaryOwnerContact ,
CIM.t$System.PrimaryOwnerName
FROM
  CIM.t$ManagedSystemElement ,
  CIM.t$System
WHERE
  CIM.t$ManagedSystemElement.id$ = CIM.t$System.id$

```

In addition to the properties of the class, the view has the following two additional fields:

- ◆ **Id\$:** An object identifier that uniquely identifies the particular instance of the class. See [“Object Identifier Id\\$” on page 795](#).
- ◆ **Class\$:** An integer field that identifies the actual type of the class. For example, the actual type of a CIM_System can be any of the concrete subclasses of CIM_System.

Views can be queried using the SELECT statement and updated using the UPDATE statement. Because views cannot be used with the INSERT and DELETE statements, use the constructor and destructor procedures.

Object Identifier Id\$

Id\$ is a 64-bit object identifier that uniquely identifies a particular instance of a class. This object identifier is usually used as an opaque handle to a particular instance. Id\$ is modeled as a signed number for ease of manipulation in Java as a long data type.

Id\$ contains the following three parts of information, which can each be extracted by invoking the appropriate stored procedure.

- ◆ The most significant 16 bits of id\$ encode the actual class of the object.
This field can be extracted using the MW_DBA.extractClass() function. This field is used for type decisions or to access additional information about the class from the MW_DBA.Class table.
- ◆ The next 8 bits of id\$ encode the site ID.
The site ID uniquely identifies the database on a particular site. This field makes the object identifier unique across as many as 256 sites so that inventory data from multiple sites can be rolled up into a single database (Root Server with Database) for querying and reporting without causing key conflicts. The site ID can be extracted using the MW_DBA.extractSite() function.
- ◆ The least significant 40 bits uniquely identify the particular instance of that class.
This part can be extracted using the MW_DBA.extractId() function. This is not useful from an end-user’s perspective.

The id\$ field is used in its entirety as an opaque handle to an instance of a class. When an association class represents a relationship between instances of two classes, the reference fields of the association hold the id\$ of the referenced instances (like the pointers). Therefore, id\$ and these reference fields are frequently used in Join conditions when constructing the database queries that reference more than one view.

Constructor

Each concrete (non-abstract) CIM class has a constructor stored procedure that must be called to create an instance of the class. This stored procedure has input parameters that allow the user to specify a value for each property in the class, and a single output parameter that returns the id\$ allocated to the created instance. The application uses this returned id\$ value to construct association classes that reference that particular instance.

The constructor is named by prefixing the root name with c\$, and each parameter is named by prefixing the root property name with p\$. For example, the constructor for CIM_UnitaryComputerSystem, a subclass of CIM_System, is named CIM.c\$UnitaryComputerSystem and is constructed for Oracle as shown in the following example:

```
CREATE PROCEDURE CIM.c$UnitaryComputerSystem
(
  p$id$ OUT NUMBER,
  p$Caption IN CIM.t$ManagedSystemElement.Caption%TYPE DEFAULT NULL,
  p$Description IN CIM.t$ManagedSystemDescription%TYPE DEFAULT NULL,
  p$InstallDate IN CIM.t$ManagedSystemElement.InstallDate%TYPE DEFAULT NULL,
  p$Status IN CIM.t$ManagedSystemElement.Status%TYPE DEFAULT NULL,
  p$CreationClassName IN CIM.t$System.CreationClassName%TYPE DEFAULT NULL,
  p$Name IN CIM.t$ManagedSystemElement.Name%TYPE DEFAULT NULL,
  p$PrimaryOwnerContact IN CIM.t$System.PrimaryOwnerContact%TYPE DEFAULT NULL,
  p$PrimaryOwnerName IN CIM.t$System.PrimaryOwnerName%TYPE DEFAULT NULL,
  p$NameFormat IN CIM.t$System.NameFormat%TYPE DEFAULT NULL,
  p$LastLoadInfo IN CIM.t$UnitaryComputerSystem.LastLoadInfo%TYPE DEFAULT
  NULL,
  p$ResetCapability IN CIM.t$UnitaryComputerSystem.ResetCapability%TYPE
  DEFAULT NULL,
  p$PowerManagementSupported IN
  CIM.t$UnitaryComputerSystem.PowerManagementSupported%TYPE DEFAULT NULL,
  p$PowerState IN CIM.t$UnitaryComputerSystem.PowerState%TYPE DEFAULT NULL
) IS
  temp NUMBER;
BEGIN
  LOOP
    SELECT CIM.s$UnitaryComputerSystem.NEXTVAL INTO temp FROM DUAL;
    SELECT MW_DBA.makeId(240, temp) INTO temp FROM DUAL;
    EXIT WHEN MOD(temp,100) != 0;
  END LOOP;
  p$id$ := temp;
```

```

INSERT INTO CIM.t$ManagedSystemElement (id$, classOid$,
Caption, Description, InstallDate, Status, Name)VALUES(p$id$,
HEXTORAW('0302100203'), p$Caption, p$Description,
p$InstallDate, p$Status, p$Name);

INSERT INTO CIM.t$System (id$, CreationClassName,
PrimaryOwnerContact, PrimaryOwnerName,
NameFormat)VALUES(p$id$, p$CreationClassName,
p$PrimaryOwnerContact, p$PrimaryOwnerName, p$NameFormat);

INSERT INTO CIM.t$UnitaryComputerSystem (id$, LastLoadInfo,
ResetCapability, PowerManagementSupported, PowerState)
VALUES (p$id$, p$LastLoadInfo,
p$ResetCapability, p$PowerManagementSupported, p$PowerState);

END;

```

Stored procedures can be called with either positional arguments or keyword arguments, or with a combination of the two. If any positional arguments are supplied, they must precede any keyword arguments. Always use keyword arguments when calling constructor stored procedures. This provides better insulation from CIM schema changes that cause either the insertion of extra parameters or the recording of existing parameters, either of which can break a positional call in a possible undetectable way. The procedures are generated such that any omitted parameters will default to NULL.

It is permissible to use the positional notation for the first parameter p\$id\$, which is the output parameter that returns the object identifier of the newly created instance.

The following code sample shows how to call a stored procedure using positional notation for the first argument and keyword notation for all subsequent arguments on Sybase.

```

CallableStatement CS =

conn.prepareCall( "{call CIM.c$UnitaryComputerSystem( ?, p$Name=?,
p$Description=?)}" )

cs.registerOutParameter ( 1, java.sql.Types.BIGINT ); //id$

cs.setString( 2, "Bogus_UCS_1" ) ; //Name

cs.setString( 3, "Created with mixture of positional & keyword args" ); //
Description

cs.executeUpdate();

long id = cs.getLong ( 1 );

SQLWarning w = cs.getWarnings();

if( w != null )

    printWarnings( w );

else

    System.out.println("Created UCS id$ = " + id );

```

The syntax for keyword notation differs in Sybase ASA and Oracle. In Sybase ASA, the syntax is **KEYWORD=value**. In Oracle, the syntax is **KEYWORD=>value**. Properly written code will dynamically construct the call string using syntax appropriate for the database in use.

Destructor

Each non-abstract CIM class has a destructor stored procedure that is called to destroy an instance of the class. This stored procedure has only one input parameter that specifies the object identifier (id\$) of the instance to be destroyed and returns no value.

The destructor deletes the appropriate rows in all relevant tables, including the rows in the inheritance chain and any associations that reference the instance being destroyed. Only the association is destroyed; the associated objects associated are not destroyed. If there is need to destroy the association, the programmers must ensure that they are not destroyed. The destructor is named by prefixing the root name with d\$ and the single object identifier parameter is named p\$id\$. This procedure is called using positional notation. For example, the destructor for CIM_UnitaryComputerSystem, a concrete subclass of CIM_System, is named as CIM.d\$UnitaryComputerSystem.

Physical Schema

The physical schema comprises elements necessary to implement the database. The physical schema differs for each database. A typical physical schema consists of:

- ◆ Table definitions 't\$xxx' Index definitions 'i\$xxx'
- ◆ Trigger definitions 'x\$xxx', 'n\$xxx' and 'u\$xxx'
- ◆ Sequence definitions (Oracle) 's\$xxx'
- ◆ Stored procedures and functions

The logical schema is layered on top of the physical schema and makes it unnecessary for users and applications to know the physical schema.

Inventory Database Schema in ZfS

The following section describes the database schema classes and the extensions and associations made to the CIM schema for use in ZfS. These extensions have ZENworks or ManageWise as their schema name. ZENworks.*classname* refers to the extended class in the ZENworks schema and ManageWise.*classname* refers to the extended class in the ManageWise schema.

The following sections will help you understand the ZfS 3 database schema:

- ◆ [“Case Study of CIM Schema Implementation in ZfS” on page 799](#)
- ◆ [“Legends for Schema Diagrams” on page 801](#)
- ◆ [“CIM Classes and Extension Classes in ZfS” on page 801](#)
- ◆ [“Schema Diagrams of CIM and the Extension Schema in ZfS” on page 803](#)
- ◆ [“Sample Inventory Database Queries” on page 808](#)

Case Study of CIM Schema Implementation in ZfS

The following scenario describes an inventoried server that has two parallel ports with a specified interrupt number.

In the following schema diagram, the CIM_UnitaryComputerSystem represents a managed inventory system.

In this illustration, class CIM.PointingDevice associates to CIM.UnitaryComputerSystem using the association CIM.SystemDevice with SystemDevice.GroupComponent pointing to CIM.UnitaryComputerSystem and SystemDevice.PartComponent pointing to CIM.PointingDevice. The relationship between the two classes is one to many. This means a computer system might have more than one pointing devices.

Class CIM.IRQ associates to CIM.PointingDevice using the association CIM.AllocatedResource. Dependent pointing to CIM.PointingDevice and Antecedent pointing to CIM.IRQ.

Class ZENworks.ZENKeyboard associates to CIM.UnitaryComputerSystem using the association CIM.SystemDevice with SystemDevice.GroupComponent pointing to CIM.UnitaryComputerSystem and SystemDevice.PartComponent pointing to ZENworks.ZENKeyboard. The relationship between the two classes is one to one. This means a computer system can have only one Keyboard.

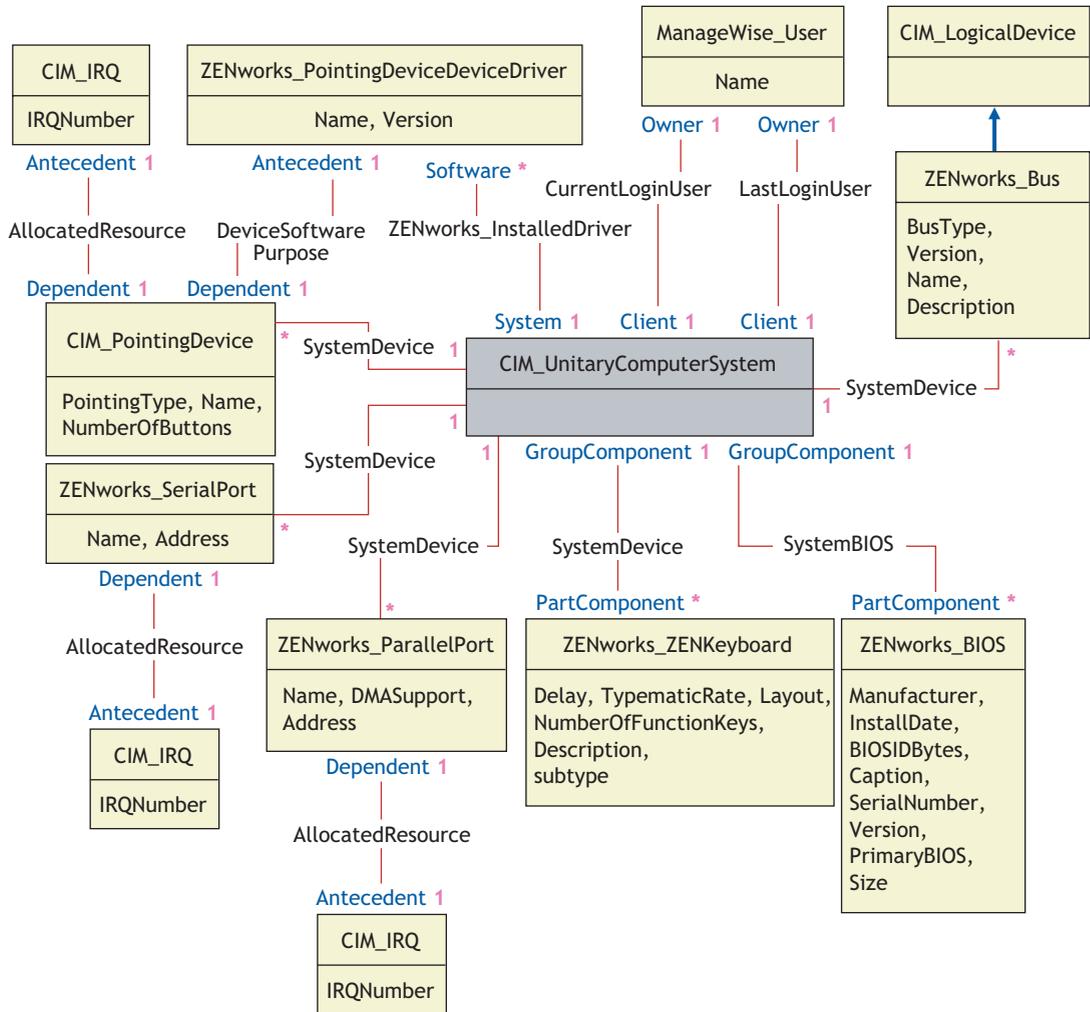
Class ZENworks.BIOS associates to CIM.UnitaryComputerSystem using the association CIM.SystemDevice with SystemDevice.GroupComponent pointing to CIM.UnitaryComputerSystem and SystemBIOS.PartComponent pointing to ZENworks.BIOS. The relationship between the two classes is one to one. This means a computer system can have only one BIOS.

Class CIM.ZENworks.ParallelPort associates to CIM.UnitaryComputerSystem using the association CIM.SystemDevice with SystemDevice.GroupComponent pointing to CIM.UnitaryComputerSystem and SystemDevice.PartComponent pointing to CIM.ZENworks.ParallelPort. The relationship between the two classes is one to many. This means a computer system might have more than one parallel port.

Class ZENworks.BUS associates to CIM.UnitaryComputerSystem using the association CIM.SystemDevice with SystemDevice.GroupComponent pointing to CIM.UnitaryComputerSystem and SystemBUS.PartComponent pointing to ZENworks.BUS. The relationship between the two classes is one to one. This means a computer system can have only one BUS.

Class ManageWise.Usera associates to CIM.UnitaryComputerSystem using CurrentLoginUser and LastLoginUser. In the CurrentLoginUser association, the specific instance of User is the one who is currently logged into the inventoried server. In the LastLoginUser association, the specific instance of User is the one who logged last into the inventoried server.

Class CIM.IRQ associates to CIM.ParallelPort using the association CIM.AllocatedResource. Dependent pointing to CIM.ParallelPort and Antecedent pointing to CIM.IRQ.



The schema diagram illustrates the following:

- ◆ All components that a computer system manages are represented as associations from the UnitaryComputerSystem class. The type of references (1..n, 1..1) between two classes are marked.
- ◆ Those associations that do not have a schema name are assumed as CIM schema.

There are three instances of ZENworks_ParallelPort associated to one instance of: CIM_UnityComputerSystem using three instances of CIM_SystemDevice associations, CIM_SystemDevice.GroupComponent references UnitaryComputerSystem, CIM_SystemDevice.PartComponent references ParallelPort.

This is called 1 to n object reference relationship and is depicted in the illustration as 1..*. Similarly, every instance of ParallelPort has a corresponding instance of CIM_IRQ designating the port's irq. This is one-to-one relationship and is depicted as 1..1.

All other classes follow similar representation. For an explanation of the CIM and extended classes, see [“CIM Classes and Extension Classes in ZfS” on page 801](#). For schema diagrams of other classes, see [“Schema Diagrams of CIM and the Extension Schema in ZfS” on page 803](#).

Legends for Schema Diagrams

The legends for reading the schema diagrams are as follows:

- ◆ Class names are enclosed in boxes with the class name as the heading and the attribute names within it.
- ◆ Red lines connect two classes using an association class.
- ◆ Blue lines indicate the class inheritance hierarchy. The class pointed by the arrow is the class that is being inherited from. The class from where the arrow emanates is the inheriting class.
- ◆ The association class name is shown within the line joining two classes.
- ◆ References of the association class are marked on either side of the associated classes.

For an explanation about CIM schema, see the CIM 2.2 schema specification on the [DMTF Web site \(http://www.dmtf.org\)](http://www.dmtf.org).

CIM Classes and Extension Classes in ZfS

The following table describes the CIM and extension classes that ZfS uses:

CIM and Extension Class in ZfS	Description of the details that the Class Models
CIM.PointingDevice	Any pointing device available on the managed system. Mostly used to model the mouse.
ZENworks.SystemInfo	Identification details about the system such as serial number and asset tag.
ZENworks.PointingDeviceDeviceDriver	Device driver that is installed with the pointing device.
ZENworks.SerialPort	Serial ports on the managed system.
ZENworks.ParallelPort	Parallel ports on the managed system.
ZENworks.ZENKeyboard	Attributes modeling the properties of the system keyboard.
ZENworks.BIOS	BIOS software on the system.
ZENworks.Bus	System bus in the system.
ManageWise.User	Details of the user who was logged in to the inventoried server.
ManageWise.MSDomainName	Name of the domain to which the Windows NT inventoried server is attached.
ManageWise.NDSName	DN name and tree under which the managed inventoried server is registered in Novell eDirectory™.
CIM.VideoBIOSElement:	Video driver.
CIM.Processor	Processor of the inventoried server.
ZENworks.Videoadapter	Properties of the monitor and the adapter connecting it.
ZENworks.ZENOperatingSystem	Details of the operating system.

CIM and Extension Class in ZfS	Description of the details that the Class Models
ZENworks.InventoryScanner	Details of the inventory scanner that has scanned for hardware and software details of the managed inventoried server.
ZENworks.NetwareClient	NetWare client version of the inventoried server.
CIM.Product	Software installed on the managed system. Key attributes are the names of the product, vendor, and version.
ZENworks.ZENNetworkAdapter	Information on the properties of the network adapter.
ZENworks.NetworkAdapterDriver	Network card adapter driver information.
CIM.IPProtocolEndpoint	IP address of the inventoried server.
CIM.IPXProtocolEndpoint	IPX address of the inventoried server.
CIM.LANEndpoint	Active MAC address.
ManageWise.DNSName	DNS name of the inventoried server.
ZENworks.SoundAdapter	Description of the multimedia adapter on the inventoried server.
ZENworks.ZENPOTSModem	Physical configuration of the modem device.
CIM.DMA	Information about the system DMA channels.
CIM.CacheMemory	Information about the configured system cache.
CIM.IRQ	List of Interrupt channels and their status on the system. They are also associated to devices that use the specified interrupt number.
ZENworks.MotherBoard	Information about the motherboard on the inventoried server.
CIM.PowerSupply	Information about the power supply unit of the inventoried server.
CIM.Battery	Physical details of the system battery.
CIM.Card	Details of adapter cards mounted on the system board.
CIM.Slot	Expansion slots available on the system board.
ZENworks.StoragePhysicalMedia	Physical information about the storage devices on the inventoried server, such as hard disk, floppy drives, CD drives, and others.
ZENworks.LogicalDiskette	Drive mapped to the floppy drive.
ZENworks.PhysicalDiskette	Derived from ZENworks.StoragePhysicalMedia to model the floppy disk drive.
ZENworks.PhysicalDiskDrive	Derived from ZENworks.StoragePhysicalMedia to model the hard disk.
ZENworks.LogicalDiskDrive	Information about the local drives on the hard disk.

CIM and Extension Class in ZfS	Description of the details that the Class Models
CIM.LocalFileSystem	Information about the local file system installed on the Windows servers.
ZENworks.PhysicalCDROM	Derived from ZENworks.StoragePhysicalMedia to model the CD drive.
ZENworks.WinOperatingSystem	Details of the Windows operating system.
ZENworks.NetWareOperatingSystem	Details of the NetWare operating system.
ZENworks.ZENDiskDrive	Details of fixed or removable disk drives.
ZENworks.LogicalCDROM	Drive mapped to the CD drive.

Schema Diagrams of CIM and the Extension Schema in ZfS

The following schema diagrams of the CIM and extension schema model the Inventory database in ZfS.

In the following schema diagram, the CIM_UnitaryComputerSystem represents a managed inventory system.

In this illustration, class CIM.PointingDevice associates to CIM.UnitaryComputerSystem using the association CIM.SystemDevice with SystemDevice.GroupComponent pointing to CIM.UnitaryComputerSystem and SystemDevice.PartComponent pointing to CIM.PointingDevice. The relationship between the two classes is one to many. This means a computer system might have more than one pointing devices.

Class CIM.IRQ associates to CIM.PointingDevice using the association CIM.AllocatedResource. Dependent pointing to CIM.PointingDevice and Antecedent pointing to CIM.IRQ.

Class ZENworks.ZENKeyboard associates to CIM.UnitaryComputerSystem using the association CIM.SystemDevice with SystemDevice.GroupComponent pointing to CIM.UnitaryComputerSystem and SystemDevice.PartComponent pointing to ZENworks.ZENKeyboard. The relationship between the two classes is one to one. This means a computer system can have only one Keyboard.

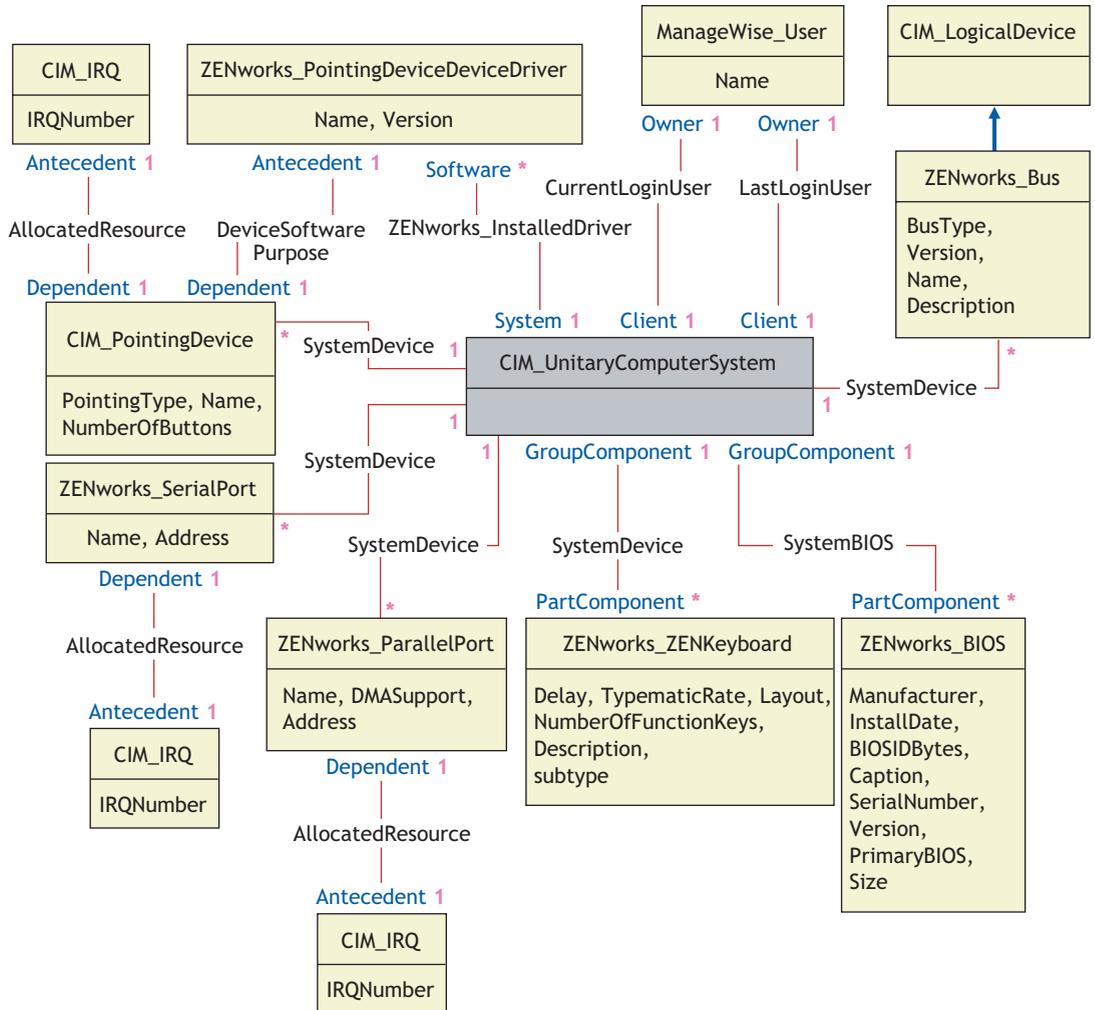
Class ZENworks.BIOS associates to CIM.UnitaryComputerSystem using the association CIM.SystemDevice with SystemDevice.GroupComponent pointing to CIM.UnitaryComputerSystem and SystemBIOS.PartComponent pointing to ZENworks.BIOS. The relationship between the two classes is one to one. This means a computer system can have only one BIOS.

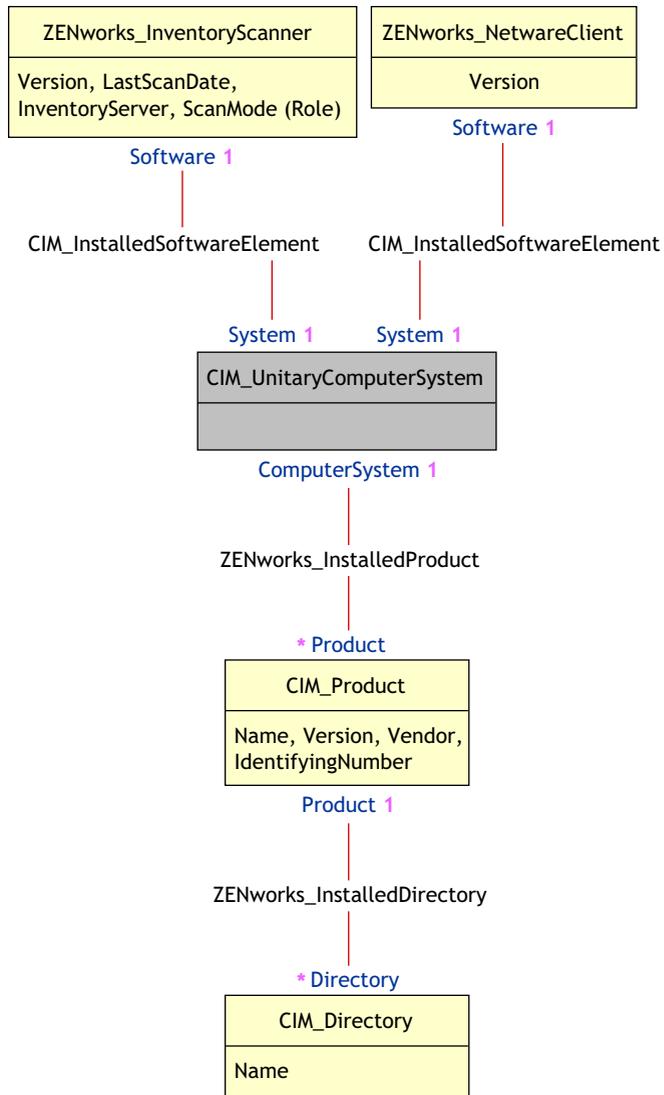
Class CIM.ZENworks.ParallelPort associates to CIM.UnitaryComputerSystem using the association CIM.SystemDevice with SystemDevice.GroupComponent pointing to CIM.UnitaryComputerSystem and SystemDevice.PartComponent pointing to CIM.ZENworks.ParallelPort. The relationship between the two classes is one to many. This means a computer system might have more than one parallel port.

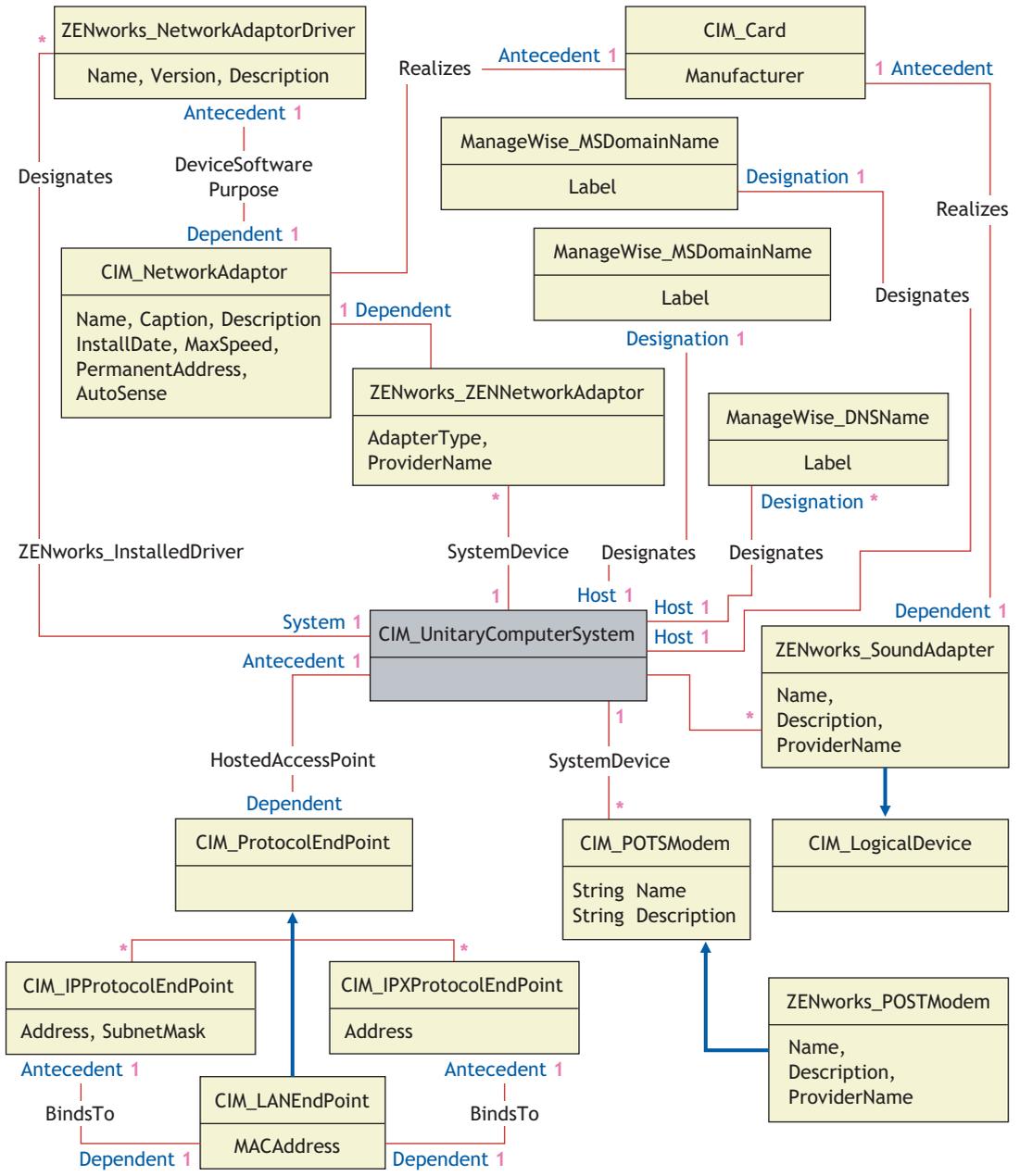
Class ZENworks.BUS associates to CIM.UnitaryComputerSystem using the association CIM.SystemDevice with SystemDevice.GroupComponent pointing to CIM.UnitaryComputerSystem and SystemBUS.PartComponent pointing to ZENworks.BUS. The relationship between the two classes is one to one. This means a computer system can have only one BUS.

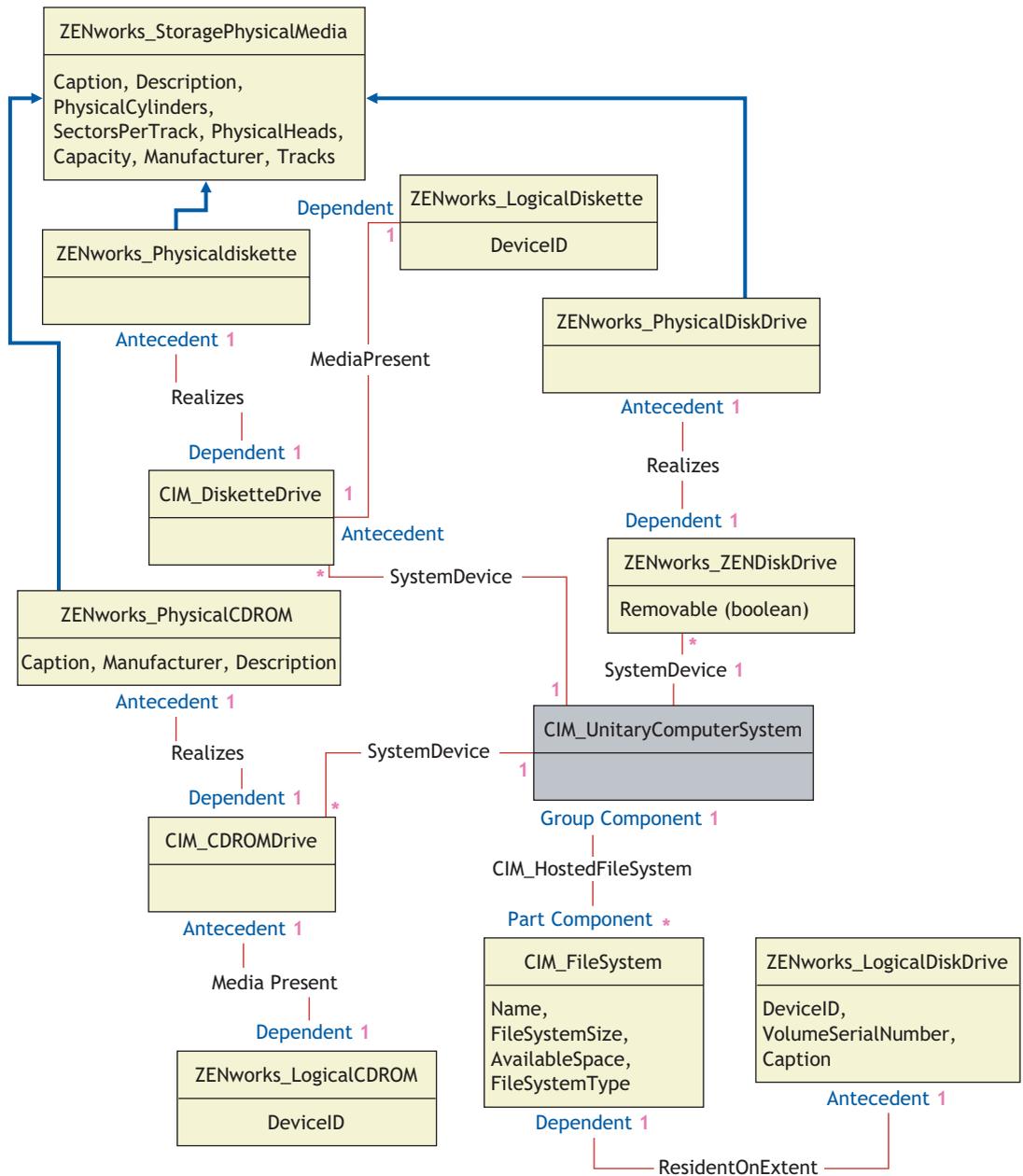
Class ManageWise.User has two associations with CIM.UnitaryComputerSystem; CurrentLoginUser and LastLoginUser. In the CurrentLoginUser association, the specific instance of User is the one who is currently logged into the inventoried server. In the LastLoginUser association, the specific instance of User is the one who logged last into the inventoried server.

Class CIM.IRQ associates to CIM.ParallelPort using the association CIM.AllocatedResource. Dependent pointing to CIM.ParallelPort and Antecedent pointing to CIM.IRQ.









Sample Inventory Database Queries

The following are sample queries for retrieving the inventory information from the ZFS Inventory database.

Refer to the schema diagrams in “[Schema Diagrams of CIM and the Extension Schema in ZFS](#)” on [page 803](#) to find out the associated schema classes and attribute information.

1. Retrieve the name and ID of all inventoried servers from the database and also to the eDirectory tree to which these servers are registered. The query is as follows:

```
SELECT u.id$,m.label,m.tree FROM managewise.NDSName
m,cim.UnitaryComputerSystem u,managewise.Designates s where
s.Designation=m.id$ and s.HOST=u.id$
```

2. Retrieve the asset tag, manufacturer, and serial number of all the inventoried servers in the database. The query is as follows:

```
SELECT m.Tag,m.Manufacturer,m.SerialNumber FROM
cim.UnitaryComputerSystem u,zenworks.SystemInfo
m,cim.ComputerSystemPackage s WHERE s.Antecedent=m.id$ and
s.Dependent=u.id$
```

3. Retrieve all the software applications with their versions that are installed on the inventoried server 'SJOHN164_99_139_79' registered under the 'NOVELL_AUS' eDirectory tree. The query is as follows:

```
SELECT m.name,m.version FROM cim.Product m,cim.UnitaryComputerSystem
u,zenworks.InstalledProduct s,managewise.NDSName
m1,managewise.Designates s1 WHERE (s.Product=m.id$ and
s.ComputerSystem=u.id$) AND (s1.Designation=m1.id$ and s1.Host=u.id$) AND
m1.label='SJOHN164_99_139_79.WS' and m1.tree='Novell_AUS'
```

4. Retrieve the processor information for the inventoried server 'SJOHN164_99_139_79'. The query is as follows:

```
SELECT
m.DeviceID,m.Family,m.Stepping,m.OtherFamilyDescription,m.MaxClockSpeed,
m.CurrentClockSpeed,m.Role,m.UpgradeMethod FROM cim.Processor
m,cim.UnitaryComputerSystem u,cim.ComputerSystemProcessor s,
managewise.NDSName m1,managewise.Designates s1 WHERE
(s.PartComponent=m.id$ and s.GroupComponent=u.id$) AND
m1.label='SJOHN164_99_139_79.WS'
```

5. Retrieve the ID of the UnitaryComputerSystem used for the inventoried server 'SJOHN164_99_139_79'. The query is as follows:

```
SELECT name from CIM.UnitaryComputerSystem WHERE
name='SJOHN164_99_139_79'
```

6. When you know the ID of the UnitaryComputerSystem for a particular inventoried server from the query as shown in query 5, query 4 can be modified as:

```
SELECT
m.DeviceID,m.Family,m.Stepping,m.OtherFamilyDescription,
m.MaxClockSpeed,m.Role,m.UpgradeMethod FROM cim.Processor
m,cim.UnitaryComputerSystem u,
cim.ComputerSystemProcessor s u.id$=? and
s.PartComponent=m.id$ and s.GroupComponent=u.id$
```

Substitute the ID of the specified inventoried server in place of the ? value for u.id in the query.

7. List the IP address, IPX address, and MAC address of all inventoried servers in the database. The query is as follows:

```
SELECT ip.Address, ipx.Address, mac.MACAddress FROM
cim.IPProtocolEndpoint ip, cim.IPXProtocolEndpoint ipx,
cim.LANEndpoint mac, cim.UnitaryComputerSystem u,
cim.HostedAccessPoint s WHERE (s.Dependent=ip.id$ and
s.Antecedent=u.id$) AND (s.Dependent=ipx.id$ and
s.Antecedent=u.id$) AND (s.Dependent=mac.id$ and
s.Antecedent=u.id$)
```

Modify the same query to get the information for a specified inventoried server as follows:

```
SELECT ip.Address, ipx.Address, mac.MACAddress FROM
cim.IPProtocolEndpoint ip, cim.IPXProtocolEndpoint ipx,
cim.LANEndpoint mac, cim.UnitaryComputerSystem u,
cim.HostedAccessPoint s WHERE (s.Dependent=ip.id$ and
s.Antecedent=u.id$) AND (s.Dependent=ipx.id$ and
s.Antecedent=u.id$) AND (s.Dependent=mac.id$ and
s.Antecedent=u.id$)AND u.id$=?
```

Use the query as shown in query 5 to retrieve the ID of the specified inventoried server and substitute the ID in place of the ? value for u.id in the query.

8. Retrieve the name and other properties of the drives on the hard disk of the specified inventoried server.

```
SELECT m.id$,n.id$,m.DeviceID,n.FileSystemSize,
n.AvailableSpace,m.VolumeSerialNumber,m.caption as
VolumeLabel, n.FileSystemType FROM
ZENworks.LogicalDiskDrive m,CIM.LocalFileSystem
n,CIM.HostedFileSystem s,CIM.ResidesOnExtent r WHERE
(s.GroupComponent=? and s.PartComponent=n.id$) AND
(r.Antecedent=m.id$and r.Dependent=n.id$)
```

Use the query shown in query 5 to retrieve the ID of the specified inventoried server and substitute the ID in place of the ? for u.id\$ in the query.

29 Managing Inventory Information

This section contains the following information to help you customize the way Novell® ZENworks® for Servers (ZfS) displays information:

- ♦ “Viewing the Inventory Servers Deployed for Inventory” on page 811
- ♦ “Viewing the Inventory Information” on page 812
- ♦ “Customizing the Inventory Information” on page 831
- ♦ “Exporting the Inventory Data to CSV Format” on page 837

Viewing the Inventory Servers Deployed for Inventory

Using ConsoleOne®, you can view the Inventory servers and databases that you configured for collecting inventory.

To get a complete Inventory tree view, you need to log into all the Novell eDirectory™ trees that contain Inventory servers present in your inventory tree.

To view the Inventory servers deployed for inventory:

- 1** In ConsoleOne, select a container, click the View menu, then click Complete Tree View.
All the Inventory servers within the container are displayed in the Complete Tree View.
To view a complete tree view if your inventory deployment involves roll-up of data between Inventory servers that are situated on different Novell eDirectory trees:
 - 1a** In ConsoleOne, select NDS Tree.
 - 1b** Click View > Complete Tree View.
 - 1c** Select the eDirectory trees or containers within the tree that contains the Inventory servers.
 - 1d** Click OK.
- 2** In ConsoleOne, right-click the Inventory Service object, click View, then click Up Tree View.
If your inventory deployment consists of a single eDirectory tree, an Up Tree View displays all the Inventory servers from the selected Inventory server up to the highest level (Root Server).
If your inventory deployment involves roll-up of inventory data across Inventory servers located on different eDirectory trees, the Up Tree View displays all the Inventory servers from the selected Inventory server up to the highest level server to which you have logged in.

Viewing the Inventory Information

The following sections will explain the various types of information you can view using ConsoleOne:

- ◆ You can list hardware and software components found on the inventoried server and any custom information you have specified for the inventoried server.

The Inventory Summary window displays the inventory items for an inventoried server. This window displays the data from the last inventory scan for the inventoried server. For more information, see [“Viewing the Inventory Summary of an Inventoried Server” on page 813](#).

- ◆ You can list inventoried servers with the inventory information from the Inventory database satisfying the criteria you specify in the Inventory Query window. You form a query by specifying the component and its attribute for servers within the selected database sites.

For more information about querying the Inventory database, see [“Viewing Inventory Information of Inventoried Servers by Querying the Database” on page 821](#).

- ◆ You can use a list of reports that generate the inventory information from the Inventory database specific to your needs.

For more information, see [“Running Inventory Reports” on page 824](#).

Configuring the Inventory Database

If you want to view the inventory information stored in the database from ConsoleOne, you must configure the database. The inventory information from the Inventory database that you configure will be used for generating inventory reports, viewing inventory information, and for querying the inventory information from the database.

To configure the Inventory database:

- 1** In ConsoleOne, select a container.
- 2** Invoke Configure DB.
 - ◆ To invoke Configure DB from a database object, right-click the database object, click ZENworks Inventory, then click Configure DB. This configures the database object.
 - ◆ To invoke the Configure DB dialog box from the ConsoleOne Tools menu, click Tools > ZENworks Inventory > Configure DB.
- 3** Click Browse to browse for and select the ZENworks Database object.

You can also select an existing ZENworks Database object from the list of Database objects.

This Database object contains the database settings such as the protocol, port in use by the database, and others.
- 4** To apply this database configuration to all the sessions, select the Apply Configuration Across Sessions check box.
- 5** Click OK.

The database you configured is used for data retrieval unless you change it again using this same procedure.

Viewing the Inventory Summary of an Inventoried Server

The Inventory Summary window displays the data from the last inventory scan for the inventoried server.

To view the inventory information of an inventoried server:

- 1 Configure the Inventory database.

For more information, see [“Configuring the Inventory Database” on page 812](#).

- 2 Right-click an inventoried server, click Actions, then click Inventory.

ZfS provides the following inventory information collected from the inventoried servers:

Scan Data Group	Scan Data Item	Description
Hardware/Software Inventory > General > System Information	Asset Tag	Asset tag number that the ROM-based setup program creates
	Computer Model	Identifying information of the computer; for example, Compaq*, Dell*, and others.
	Computer Type	Type of computer, such as IBM* PC, and others
	Machine Name	DNS name of the inventoried server
	Management Technology	Technology available on the inventoried server such as DMI, WMI, and others
	Model Number	Serial number value for the computer, assigned during manufacture
Hardware/Software Inventory > General > System Identification	Primary Owner Name	The name of the primary user or owner of this system
	Primary Owner Contact	The phone number of the primary user of this system
	Name	Name of the inventoried server as represented in eDirectory such as the fully qualified DN of the inventoried server
Hardware/Software Inventory > General > Login Details > Windows Domain	Windows NT Domain Name	Domain name of the inventoried server
Hardware/Software Inventory > Software > Software Vendors	Name	Name of the software manufacturer
	Version	Version number of the software
	Identification Number	Product ID
	Location	Installation directory
Hardware/Software Inventory > Software > Device Drivers > Pointing Drivers >	Name	Name of the mouse driver
	Version	Version number of the mouse driver

Scan Data Group	Scan Data Item	Description
Hardware/Software Inventory > Software > Device Drivers > Display Drivers	Install Date	Install date of the display driver
	Manufacturer	Name of the display driver manufacturer
	Is Shadowed (True or False)	If True, the display driver is currently being shadowed
	Version	Version number of the display driver
Hardware/Software Inventory > Software > Device Drivers > Network Drivers	Description	Description of the network driver
	Name	Network driver name
	Version	Version number of the network driver
Hardware/Software Inventory > Software > Operating System	Code Page	Language code page of the operating system
	OS Type	Operating system of the inventoried servers
	Install Date	Install date of the operating system
	Caption	Operating system name. For example, Windows* 95/ Windows 2000
	Other Description	Page file size
	Role	Type of the operating system, such as server
	Total Virtual Memory Size	Total number of bytes in the virtual address space of the calling process
	Total Memory Size	Total memory of the operating system
Version	Version number of the operating system	
Hardware/Software Inventory > Software > NetWare Client Mode	NetWare Client Version	Version number of the NetWare® client
Hardware/Software Inventory > Software > Inventory Scanner Information	Inventory Server	Name of the inventory server to which the scans are sent
	Scan Mode	Mode used by the Scanner to scan the inventoried server
	Version	Version number of the Scanner
Hardware/Software Inventory > Hardware > Pointing Device >	IRQ Number	Interrupt assigned to this device
	Name	Identifying information of the mouse
	Number of Buttons	Number of buttons on the mouse

Scan Data Group	Scan Data Item	Description
Hardware/Software Inventory > Hardware > Keyboard	Delay	Delay before the repeat of a key
	Description	Description of the keyboard, such as IBM Enhanced 101 or 102 keys
	Layout	Layout of the keyboard
	Number of Function Keys	Total number of function keys
	Subtype	Type of the keyboard
	Typematic Rate	Rate of processing the keys
Hardware/Software Inventory > Hardware > Memory	Total Memory	Total memory of the inventoried servers
Hardware/Software Inventory > Hardware > Display Adapter	Chip Set	Chip set used by the controller to compare stem capabilities
	Current Bits/Pixel	Number of adjacent color bits for each pixel
	Current Horizontal Resolution	Number of horizontal pixels shown by the display
	Current Vertical Resolution	Number of vertical pixels shown by the display
	DAC Type	Digital-to-Analog converter type
	Description	Description of the display adapter
	Maximum Memory Supported	Maximum memory that the display adapter supports for VIDEO RAM
	Maximum Refresh Rate	Maximum refresh rate of the monitor for redrawing the display, measured in Hertz
	Minimum Refresh Rate	Minimum refresh rate of the monitor for redrawing the display, measured in Hertz
	Number of Color Planes	Number of color planes supported by the video system
	Provider	Vendor name
	Video Architecture	The architecture of the video subsystem in this system, for example, CGA/VGA/SVGA/8514A
	Video Memory Type	The type of video memory for this adapter, for example, VRAM/SRAM/DRAM/EDO RAM

Scan Data Group	Scan Data Item	Description
Hardware/Software Inventory > Hardware > BIOS	BIOS Identification Bytes	Bytes in the BIOS that indicates the computer model
	Install Date	The manufacturing date of the BIOS
	Manufacturer	BIOS vendor name
	Caption	BIOS label
	Primary BIOS	True state indicates Primary BIOS
	Serial Number	Serial number of the computer, assigned during manufacture
	Size	Size of the BIOS
	Version	Version or revision level of the BIOS
Hardware/Software Inventory > Hardware > Processor	Current Clock Speed (in MHz)	Current clock speed of the processor
	Device ID	Special hexadecimal string identifying the processor type
	Maximum Clock Speed (in MHz)	Maximum clock speed of the processor
	Other Family Description	Additional description about the Processor Family, such as Pentium* Processor with MMX technology
	Processor Family	Identification of the processor family such as Pentium II, Pentium III, and others
	Processor Stepping	Single-byte code characteristic provided by microprocessor vendors to identify the processor model
	Role	Type of processor such as central processor, math coprocessor, and others
	Upgrade Method	The method by which this processor can be upgraded, if upgrades are supported
Hardware/Software Inventory > Hardware > Modem	Description	Additional information about the modem
	Name	Identifying information of the modem
	Device ID	Special hexadecimal string identifying the modem type
	Provider	Name of the vendor

Scan Data Group	Scan Data Item	Description
Hardware/Software Inventory > Hardware > Battery	Chemistry	The battery chemistry, for example, lithium-ion or nickel metal hydride
	Design Capacity	The design capacity of the battery in mWatt-hours
	Design Voltage	The design voltage of the battery in mVolts
	Install Date	The battery manufacture date
	Manufacturer	The name of the company that manufactured the battery
	Name	Device name for this battery, for example, Duracell* DR-36
	Serial Number	The serial number for this battery
	Smart Battery Version	The Smart Battery Data Specification version number supported by this battery
Hardware/Software Inventory > Hardware > Power Supply	Description	Expanded description of the input voltage capability for this power supply
	Total Output Power (in MilliWatts)	Attribute value that represents the total output power of the power supply
Hardware/Software Inventory > Hardware > Disk Drives > Floppy	Capacity	Floppy drive capacity
	Description	Floppy drive description
	Drive Letter	Letter name of the drive
	Manufacturer	Vendor name
	Physical Cylinders	Floppy drive cylinders
	Physical Heads	Floppy drive R/W heads
	Sectors/Track	Floppy drive sectors per track
Hardware/Software Inventory > Hardware > Disk Drives > Physical Disk > Fixed Disk	Description	Description
	Manufacturer	Vendor name
	Physical Cylinders	Number of cylinders
	Physical Heads	Number of heads
	Sectors/Track	Fixed disk drive sectors per track
	Size	Size of the fixed disk

Scan Data Group	Scan Data Item	Description
Hardware/Software Inventory > Hardware > Disk Drives > Physical Disk > Removable Disk	Description	Description
	Manufacturer	Vendor name
	Physical Cylinders	Number of cylinders
	Physical Heads	Number of heads
	Sectors/Track	Fixed disk drive sectors per track
	Size	Size of the removable disk
Hardware/Software Inventory > Hardware > Disk Drives > Hard Disk > Logical Disk	Drive Letter	Letter name of the drive
	File System Type	Type of File System such as File Allocation Table (FAT)
	Free Size	Drive's actual size in MB
	Volume Label	Name of the hard disk volume
	Size	Drive's available space in MB
	Volume Serial Number	Hard disk volume serial number
Hardware/Software Inventory > Hardware > Disk Drives > CDROM	Name	Name of the CD drive attached to the inventoried servers
	Description	Description of the CD drive
	Drive Letter	Mapped drive name of the CD drive
	Manufacturer	Vendor Name
	Caption	CD's caption name
Hardware/Software Inventory > Hardware > Ports > Serial Port	Address	Base input-output address for this serial port
	IRQ Number	IRQ number of the serial port
	Name	The logical name of the I/O device on this serial port, under this operating environment
Hardware/Software Inventory > Hardware > Ports > Parallel Port	Address	Base I/O address for this parallel port
	DMA Support (True or False)	If True, DMA is supported
	Name	The logical name of the input-output device on this parallel port, under this operating environment
	IRQ Number	IRQ number of the parallel port

Scan Data Group	Scan Data Item	Description
Hardware/Software Inventory > Hardware > Bus	Bus Type	Bus type indicates PCI, ISA, and others
	Description	Bus description
	Name	Bus name
	Device ID	Special hexadecimal string identifying the bus type
	Version	Version of the bus supported by the motherboard
Hardware/Software Inventory > Hardware > Network Adapter	Adapter Type	Types of network adapter such as FDDI, token ring, etc.
	Auto Sense	A Boolean value indicating whether the network adapter is capable of automatically determining the speed or other communication characteristics of the attached network media
	Card Manufacturer	Name of the card manufacturer
	Description	Adapter description
	Install Date	Install date of the network adapter
	Maximum Speed	Rate at which the data is transferred over the LAN
	Name	Network adapter name
	Permanent Address	Node address stored permanently in the adapter
	Provider	Name of the provider
Hardware/Software Inventory > Hardware > Sound Adapter	Description	Description of the multimedia component for the inventoried server
	Name	Label of the multimedia card
	Provider	Name of the provider
Hardware/Software Inventory > Network > DNS	DNS Name	The DNS name of the inventoried server
Hardware/Software Inventory > Network > Network (<i>instance_number</i>) > IP	IP Address	The unique address assigned to a computer on an IP Internet
	Subnet Mask	The subnet mask of the inventoried server paired with an IP address specifies to an IP router which octets or bits in the IP address are the network ID and which octets or bits are the node ID
Hardware/Software Inventory > Network > Network (<i>instance_number</i>) > IPX	IPX Address	The IPX™ address of the inventoried server
Hardware/Software Inventory > Network > Network (<i>instance_number</i>) > MAC	MAC Address	Unique node address permanently coded in the network adapter that identifies a specific computer on a network

Scan Data Group	Scan Data Item	Description
Hardware/Software Inventory > Network > IP	IP Address	The unique address assigned to a computer on an IP Internet
	Subnet Mask	The subnet mask of the inventoried server paired with an IP address specifies to an IP router which octets or bits in the IP address are the network ID and which octets or bits are the node ID
Hardware/Software Inventory > Network > IPX	IPX Address	The IPX address of the inventoried server
Hardware/Software Inventory > Network > MAC	MAC Address	Unique node address permanently coded in the network adapter that identifies a specific computer on a network
Hardware/Software Inventory > System > System IRQ	Availability	Availability of the specific IRQ channel
	IRQ Number	Number of the Interrupt Request Line (IRQ), from 0 to 15
	IRQ Trigger Type	IRQ Trigger type
	Shareable	If True, the system IRQ can be shared across devices
Hardware/Software Inventory > System > System Cache	Associativity	Defines the system cache associativity (direct-mapped, 2-way, 4-way)
	Cache Type	Defines the system cache type, for example, Instruction, Data, Unified
	Capacity	Size of the data store where the cache information is kept
	Error Methodology	Error correction scheme supported by this cache component, for example, Parity/Single Bit ECC/MultiBit ECC
	Level	Indicates the cache level; internal cache that is built in to the microprocessors; external cache that is between the CPU and DRAM
	Line Size	Size in bytes of a single cache bucket or line
	Read Policy	Indicates whether the data cache is for read operation
	Replacement Policy	Algorithm that the cache uses to determine which cache lines or buckets should be reused
	Speed	Speed of this System Cache module in nanoseconds
	Write Policy	Indicates the two different ways (Write-Back and Write-Through Cache) that the cache can handle to write to the memory

Scan Data Group	Scan Data Item	Description
Hardware/Software Inventory > System > System DMA	Availability	Indicates whether Virtual Direct Memory Access (DMA) is supported
	Description	Name of the logical device that is currently using this DMA channel
	DMA Burst Mode	A data transmission mode in which data is sent faster than normal
	DMA Channel Number	Number of the Direct Memory Access (DMA) channel that a computer uses for transferring data to and from devices quicker than from computers without a DMA channel
Hardware/Software Inventory > System > System Slot	Description	Card currently occupying this slot
	Maximum Data Width	Maximum bus width of cards accepted in the slot
	Thermal Rating	Maximum thermal dissipation of the slot in milliwatts
Hardware/Software Inventory > System > Motherboard	Manufacturer	Name of the motherboard manufacturer
	Number of Slots	The number of expansion slots in the motherboard for adding more memory, graphic capabilities, and support for special devices
	Version	Version of the motherboard
	Description	General description of the motherboard

NOTE: For an enumerated attribute, the value will be displayed in the format *enumerated_value [enumerated_ID]*. For example, Processor.Processor Family = Pentium (R) III [17].

The Status bar displays the following information:

- ◆ **Tree Name:** Displays the eDirectory tree name where the inventoried workstation or inventoried server resides.
- ◆ **Recent Information:** Set to Yes if the Inventory database has been updated with the latest inventory information of the selected inventoried server.

Viewing Inventory Information of Inventoried Servers by Querying the Database

Using ConsoleOne, you can query the Inventory database to display the hardware and software components of inventoried servers that you want to view. The Inventory Query window displays the information satisfying the criteria you specify.

The Inventory database stores inventory data (general, hardware, software, network, and system information) for each inventoried server. Querying the Inventory database helps to create groups of similar devices and to focus your reports on specific types of machines. For example, you can query the database to find machines that have an i486D processor and a VGA card.

To query the Inventory database for inventory information:

- 1 In ConsoleOne, click a container.

2 Invoke Query.

- ◆ To invoke the Inventory query from a database object, right-click the database object, click ZENworks Inventory, then click Inventory Query.
- ◆ To invoke the Inventory query from the ConsoleOne Tools menu, you must first configure the database and then click Tools > ZENworks Inventory > Inventory Query. For more information on how to configure the Inventory database, see [“Configuring the Inventory Database” on page 812](#).

3 Specify the criteria for query:

Query the Inventory database for: By default, the Servers option will be enabled. The query locates all inventoried servers satisfying the query expression. If ZENworks for Servers 3 and ZENworks for Desktops 4 are installed in the same environment; the Workstations, the Servers and the Both options will be available. When you select Servers, the query locates all inventoried servers satisfying the query expression. Choose Both to include all workstations and inventoried servers satisfying the query expression.

Find Type: Select Quick or Advanced. Click Quick to specify a simple query. When you choose a Quick query, you specify one attribute, relational operators, and the value of the attribute. Choose Advanced query to specify many attributes. You can combine multiple query groups so each group defines a set of query criteria. For example, use the Advanced query to run a query to discover all devices in the database with 486 processors and use query connectors, and add another query to discover which of these inventoried servers have a VGA color video adapter.

Display Machine(s) Not Satisfying the Query: Select the check box to retrieve machines that do not satisfy the query.

Select Attribute: Select the component or component attributes. Attributes that you can specify to query on the inventoried servers are grouped into the following categories: General, Software, Hardware, Network, and System.

The custom attribute will be prefixed by an asterisk (*).

For example, to find the machines that do not have pointing device installed, select Pointing Device as the component. To specify the version of BIOS as a component in the query, select BIOS as the component and VERSION as the component attribute.

Operator or Relational Operator: Select to determine the relationship between the components and the value. The relational operators are grouped on the basis on the data type of the attribute selected in the Select Attribute window as shown in the following table:

Data Type of the Attribute	Relational Operators
String	Equal To (=), Not Equal To (!=), Matches ([]), Does Not Match (![]) and Is NULL (null)
Numeric	Equal (=), Not Equal (!=), Less Than (<), Less Than or Equal To (<=), Greater Than (>), Greater Than or Equal To (>=), and Is NULL (null)
Date	On (=), After (>), On or After (>=), Before (<), On or Before (<=), and Is NULL (null)
Enum	Equal To (=), Not Equal To (!=), and Is NULL (null)
Custom	Includes all the relational operators that are grouped under the String, Numeric, and Date data types

NOTE: If the query does not display the result when the data type of the attribute is Custom and the relational operator is Numeric or Date, use the Equal To operator to find the values for the custom attributes that are stored in the Inventory database.

If you select only the component in the Select Attribute window, the Relational Operator will be set to NULL by default and other relational operators will not be available.

Value: Description values are the possible values of an inventory component. For example, 6.0 is a possible value for the DOS-Version attribute. Description values are not case sensitive.

NOTE: For an enumerated attribute, the value will be displayed in the format, *enumerated_value* [*enumerated_ID*]. For example, Processor.Processor Family = Pentium (R) III [17].

If you choose Matches ([]) or Does Not Match (![]) as the relational operator, you can use wildcards to substitute characters in the Value field. The following table lists the wildcards that can be used according the SQL documentation:

Example	Specifies to Include
?	Any one character
_ (underscore)	Any one character
%	Any string of zero or more characters
[]	Any one character in the specified range or set
[^]	Any one character not in the specified range or set

NOTE: To define a query using special characters such as ? or [, specify the query in the following formats: [?] or [[]].

The list of description values displayed for an Inventory component is taken from the Inventory database corresponding to the component.

Logical Operator: This option is available only for the Advanced query. Logical Operator forms query groups that will be combined with the previous query group by using the relational operator specified between the query groups.

Save: This option is available only for the Advanced query. It saves the query expression as a file in the location that you specify. The query file does not have a default extension; however, we recommend the .QRY extension for easy reference.

Load: This option is available only for the Advanced query. It loads the query file that you specify. You must provide the full filename with its extension.

4 Click Find.

This will query based on the query criteria you specify and display the inventoried servers that match the query in the Query Results window.

In the Query Results window, double-click the inventoried server or click File > Advanced Query to view the **inventory information** of the inventoried server.

Usage of Relational Operators

- ◆ **Match:** Use the Match operator to find the inventoried servers that satisfy the query condition.
For example, use the Match operator to find all the inventoried servers with IP address 164.99.151.%,
- ◆ **NULL:** Use the NULL operator to query for those inventoried servers whose particular attribute is not scanned but the component has been scanned and some attributes are populated.
For example, to find a list of inventoried servers for which BIOS.Manufacturer is not scanned, form a BIOS.Manufacturer is NULL query. This query will display the inventoried servers for which the BIOS has been scanned.
- ◆ **NOT SATISFYING:** Use the NOT SATISFYING query (or the NOT SATISFYING filter condition) to find filter conditions for the inventoried servers that negate the given query.
For example, two servers S1 and S2 contain serial ports COM1 and COM2. The query (SerialPort='COM1') will return S1 and the query (SerialPort!='COM1') will also return the S1 because S1 contains the serial port COM2. To query the inventoried servers that do not contain the serial port COM1 you must use <NOT SATISFYING>(SerialPort='COM1'). To use the NOT SATISFYING option, click the Display Machines Not Satisfying the Query check box in the query window.

Running Inventory Reports

You can run reports to gather inventory information from the Inventory database. The Inventory reports are designed using Crystal Reports.

You can select from a predefined set of report forms to generate a report. The inventory report is displayed in the Crystal Viewer window.

You can print or export the report as desired. Remember that any reports you generate will be empty if you have not configured ZfS to start populating the Inventory database with the data you want.

This section covers information on the following sections:

- ◆ [“Prerequisites for Generating Inventory Reports” on page 824](#)
- ◆ [“Types of Inventory Reports” on page 825](#)
- ◆ [“Generating Inventory Reports” on page 827](#)
- ◆ [“Printing an Inventory Report” on page 829](#)
- ◆ [“Exporting an Inventory Report to a File” on page 829](#)
- ◆ [“Running Inventory Reports” on page 824](#)
- ◆ [“Understanding User-Defined Reports” on page 829](#)

Prerequisites for Generating Inventory Reports

Before running the inventory reports you must make sure that the appropriate ODBC client for Sybase* or Oracle* is installed on the machine running ConsoleOne. The ODBC driver will be automatically configured on the machine when you invoke the Inventory report.

You can install the Sybase ODBC driver version 7.0.0.313 from the *ZENworks for Servers Companion CD*. To install the Sybase ODBC driver, copy the \odbc\sybase\sybaseodbc.zip from

the *ZENworks for Servers Companion CD* to a drive. For installation instructions, refer to the `odbc\sybase\odbcreadme.txt` on the *ZENworks for Servers Companion CD*.

For Oracle, you must install Oracle client only for ODBC because Inventory reports are not compatible with either the older or the later version of the client.

Types of Inventory Reports

You can generate the types of reports described below, assuming you have already configured ZFS to start populating the inventory database with the data you want. The following table gives the Simple Inventory lists that provide information on individual aspects of Server Inventory, such as the operating system and the selection criteria. The table also lists the Comprehensive Inventory Reports that combine several aspects of Server Inventory into each report, such as memory, hard disk, and processor.

Inventory Report Group	Report Name	Selection Criteria	Information Displayed in the Inventory Report
Hardware Inventory	Asset Management Report	Scope, Distinguished Name, Distinguished Tree Name, IP Address, and DNS Name You can also select to display the following options in the report: Memory, Processor, Display Adapter, Keyboard, Pointing Device, Fixed and Removable Disk, Floppy, CD ROM, and Network Adapter.	Memory, processor, display details, keyboard, pointing device, fixed and removable disk, floppy, CD drive, and network adapter details for each system
	BIOS Listing	Scope, Distinguished Name, Distinguished Tree Name, IP Address, DNS Name, BIOS Install Date, and Manufacturer	List of all the machines with a BIOS manufacturer, BIOS release date, and the total number of such machines
	Devices Listing	Scope, Distinguished Name, Distinguished Tree Name, IP Address, DNS Name, and Devices Based on the device selected in the Devices drop-down list, the filter condition for the selected device will be displayed.	List of all machines with a particular device. The devices are pointing device, keyboard, bus, video adapter, network adapter, sound adapter, modem, battery, and power supply.
	Storage Devices Inventory Report	Scope, Distinguished Name, Distinguished Tree Name, IP Address, and DNS Name You can also select to display the following options in the report: Fixed disk and Removable Disk, Logical Disk, Floppy, and CD ROM.	Fixed disk, removable disk, logical disk, floppy, and CD drive details for each system
	Storage Device Listing	Scope, Distinguished Name, Distinguished Tree Name, IP Address, DNS Name, and Devices Based on the storage device selected in the Devices drop-down list, the filter condition for the selected device will be displayed.	List of all machines with a particular storage device. The storage devices are fixed and removable disk, floppy, and CD drive
	System Information Listing	Scope, Distinguished Name, Distinguished Tree Name, IP Address, and DNS Name	List of all machines with system information for each machine

Inventory Report Group	Report Name	Selection Criteria	Information Displayed in the Inventory Report
System Configuration Inventory	Hardware Summary Report	Scope, Distinguished Name, Distinguished Tree Name, IP Address, DNS Name, Operating System Type, Operating System Version, Processor Family, Max Clock Speed (Lower Bound in MHz), Max Clock Speed (Upper Bound in MHz), Total Memory (Lower Bound in MB), Total Memory (Upper Bound in MB), Fixed disk Size (Lower Bound in GB), and Fixed Disk Size (Upper Bound in GB)	Operating system name, operating system version, processor family, processor maximum clock speed, memory, and fixed disk size for each machine
	Memory Listing	Show Chart, Scope, Distinguished Name, Distinguished Tree Name, IP Address, DNS Name, Total Memory (Lower Bound in MB), and Total Memory (Upper Bound in MB)	List of all the machines within a range of memory size (such as 200-400 MB) and the total number of such machines
	Networking Information Report	Scope, Distinguished Name, Distinguished Tree Name, IP Address, and DNS Name You can also select to display the following options in the report: Network Adapter Type, DNS Name, IP Address, MAC Address, IPX Address, and Windows Domain Name.	Network adapter type, DNS, IP address, MAC address, IPX address, and Windows Domain name for each system
	Operating System Listing	Show Chart, Scope, Distinguished Name, Distinguished Tree Name, IP Address, DNS Name, Operating System Type, and Operating System Version	List of all the machines with an operating system type, an operating system version, and the total number of such machines
	Processor Listing	Show Chart, Scope, Distinguished Name, Distinguished Tree Name, IP Address, DNS Name, Processor Family, Maximum Speed (Lower Bound in MHz), Maximum Speed (Upper Bound in MHz), Current Speed (Lower Bound in MHz), and Current Speed (Upper Bound in MHz)	List of all the machines with a processor family (such as Pentium Pro), processor maximum clock speed, and processor current clock speed of such machines
	System Internal Hardware Inventory Report	Scope, Distinguished Name, Distinguished Tree Name, IP Address, and DNS Name You can also select to display the following options in the report: System IRQ, System Cache, System DMA, System Slot, and Motherboard.	IRQ, cache, DMA, slot, and motherboard for each system
Software Inventory	Application Software Inventory Report	Product Location, Scope, Distinguished Name, Distinguished Tree Name, IP Address, DNS Name, Include Product Location, Software Vendor, Software Name, and Software Version	Software with product name, version, vendor, product ID, product location, and recent information for each system
	Software Listing	Include Product Location, Scope, Distinguished Name, Distinguished Tree Name, IP Address, DNS Name, Software Vendor, Software Name, and Software Version	List of all the machines with a software vendor, software name, version, and the total number of such machines

Inventory Report Group	Report Name	Selection Criteria	Information Displayed in the Inventory Report
	Software Summary Listing	Show Chart, Scope, Software Vendor, Software Name, and Software Version	Lists the number of machines with a particular software version TIP: The Software Summary Listing chart might not be displayed properly because there is too much software data in your Inventory database. For the chart to be displayed properly, use the selection criteria effectively to restrict the results displayed according to your requirement.
	System Software Inventory Report	Scope, Distinguished Name, Distinguished Tree Name, IP Address, and DNS Name You can also select to display the following options in the report: Display Driver, Pointing Device Driver, Network Adapter Driver, and NetWare Client.	Drivers (such as pointing device drivers, network adapter drivers, and display drivers) and NetWare Client for each system.
Others	Inventory Scan Listing	Show Chart, Scope, Distinguished Name, Distinguished Tree Name, IP Address, DNS Name, Last Scan Date (On or Before), Inventory Server Name, and Recent Information	Date and time of the last inventory scan, Inventory server name, and recent information on each system
	User Defined Reports For more information on how to create user-defined reports, see “Understanding User-Defined Reports” on page 829	Based on the options specified by the user in the CONSOLEONE\ConsoleOne_version\BIN\USERREPORTS.INI file	Displays the user-defined report.

NOTE: The Show Chart selection criteria display a graphical representation of the Inventory report.

Generating Inventory Reports

To generate the inventory report:

- 1 Invoke the Inventory report by using any of the following methods:
 - ◆ To invoke the Inventory report from a database object, right-click the database object, then click ZENworks Reports
 - ◆ To invoke the Inventory report from the ConsoleOne Tools menu, you must first configure the database (click Tools > ZENworks Inventory > Configure DB), then click Tools > ZENworks Reports.

2 Click the report you want to generate.

The description for the report is displayed on the right side of the screen.

See the table with listing of simple Inventory lists and listing of the comprehensive inventory reports.

3 Specify the selection criteria.

The Scope selection criteria are enabled only if both Zfd 4 and Zfs 3 are installed on the same machine.

For example, if you want to view the inventory information of all inventoried servers, select Server as the scope selection criteria. The report will display the inventory information of all servers within the configured Inventory database.

Depending on the type of report you want, you can filter the information. For example, to view all inventoried servers with the Windows NT* operating system, you select the Operating System Listing, and specify the selection criteria Scope as Both, the Operating System Type as Windows NT, and the Operating System Version as 3.0.

Follow these guidelines as you work with the Reporting dialog box:

- ◆ The selection criteria in the Inventory report are case sensitive.
For example, if you want to know the list of machines whose Distinguished Name is CN=MACHINE1.OU=ENG.O=NOVELL, specify OU=ENG.O=NOVELL as the selection criterion. All the machines whose DN contains OU=ENG.O=NOVELL are displayed in the Inventory report, but the machines whose DN contains ou=eng.o=novell are not displayed in the Inventory report.
- ◆ If the Reporting dialog box allows wildcards, you can use an asterisk (*) or question mark (?) with all selection criteria except for Distinguished Name and Distinguished Tree Name. The wildcard characters can be used for character data only.

For Distinguished Name or Distinguished Tree Name, if you specify only a part of the DN, all machine names containing the specified string in the DN are displayed. For example, if you want to know the list of machine whose Distinguished Name contains novell.invtree, specify novell.invtree as the selection criterion; all the machines whose DN contains novell.invtree are displayed in the Inventory report.

The following table lists examples of wildcards usage.

Example	Specifies to Include
*	All items
164.99.*	All items starting with 164.99.
164.9?.215.23	All items starting with 164.9, followed by any character, and ending with ".215.23"
164.96.215.23	The single named item, in this case the inventoried server with the specified IP address

4 Click Run Selected Report.

A status box appears displaying the progress of the report generation. When the report is generated, it appears in the viewer. Use the buttons on the toolbar to page through, print, or export the report.

Printing an Inventory Report

To print a report:

- 1** Generate and view the report.
- 2** To change the default settings of the Printer, click the Printer Setup icon  and modify the settings.
- 3** Click the Printer icon .

Exporting an Inventory Report to a File

To export an inventory report to a file:

- 1** Generate and view the report.
- 2** On the toolbar, click the Export Report icon .
- 3** In the Export dialog box, specify the location and file format.

If you choose to export the Inventory report to a text file, in the Export to Text dialog box, select the User defined option and set the value to 16 because the data exported will be truncated if the value is less than 16.

If you want to export the Inventory report to an HTML file, you can select HTML 3.2 or HTML 4.0 (DHTML) file format. We recommend that you export to HTML 4.0 (DHTML) because the data exported to HTML 3.2 will not be formatted properly.

If you want to export the Inventory report to a comma-separated value (.CSV) file, do the following:

- 3a** Export the report to Microsoft* Excel.
 - NOTE:** If you choose to export to .CSV, the report will not be properly exported.
- 3b** Open the .XLS file.
- 3c** Click File > Save As.
- 3d** In the Save as type field, choose CSV (Comma delimited) (*.csv).
- 3e** Click Save.
- 4** Click OK.
- 5** Browse for and select the directory where you want to save the exported file.
- 6** Click OK.

Understanding User-Defined Reports

Using the Crystal Report Designer you can generate reports with the data present in the Inventory database.

Before generating the reports, you must ensure that the report file (.rpt) is created using Crystal Report Designer 8.0 or later. For more information on how to create a .RPT file, see the Crystal Report documentation.

To generate the User-defined Inventory report:

- 1** On the machine where you are designing the report, set the ODBC DSN name to ZenInventory.

To set the ODBC name:

- 1a** Click Start > Settings > Control Panel > ODBC Data Sources (32 Bit) > Add.
- 1b** Select the ODBC driver for the database you want to connect to.
- 1c** Click Finish.
- 1d** Specify the Data Source name as ZenInventory and specify the details.

NOTE: If you want to specify a data source name other than ZenInventory, you must configure the ODBC name on the each of the machines where you invoke user-defined reports through ConsoleOne.

- 2** After you have designed the report, place the report in the
\\consoleone\version\reporting\canned\novellreporting\zeninventory\en directory.
- 3** Set the values in the userreports.ini file in the \\consoleone\version\bin directory. The userreports.ini file must contain the following values:

```
#[ReportName] <actual name of the report file without the .rpt extension>
#DisplayName=User Defined Report's display name
#Param1=Constant,Display name,<if combo then {val-1|val-2|val-3}>
#<where Param1 is the internal name of the parameter as stored in the .rpt
file>
#<Constants are 1, 2 and 3 for Combo selection, text field and numeric
field respectively>
```

For example, you can set the value as given below:

```
[ListSystemInformation]DisplayName=System Information
Role=1,Role,{2|3|5}
IPAddress=2,IP Address
DNName=2,Distinguished Name
DNTree=2,Distinguished Tree
DNSName=2,DNS Name

[ListMemory]
DisplayName=Memory
Role=1,Role,{2|3|5}
IPAddress=2,IP Address
DNName=2,Distinguished Name
DNTree=2,Distinguished Tree
DNSName=2,DNS Name
MemoryLowerLimit=3,Memory Lower Bound
```

- 4 After you set the values in the userreports.ini file, the User Defined Report is displayed in the Inventory Reports tree. You can specify multiple reports in the userreports.ini files.

NOTE: If the userreports.ini file is empty, the user cannot view the User Defined Reports in the Inventory Reports tree.

- 5 Click Run Selected Report.

Customizing the Inventory Information

This section describes how to customize the inventory information.

- ◆ [“Customizing Software Scanning of Inventoried Servers” on page 831](#)
- ◆ [“Scanning for Vendor-Specific Asset Information from DMI” on page 833](#)
- ◆ [“Customizing the Software Scanning Information of Vendors and Products” on page 835](#)
- ◆ [“Customizing the Hardware Scanning Information of Jaz and Zip Drive Vendors” on page 837](#)

Customizing Software Scanning of Inventoried Servers

You can customize the list of software applications that you want to scan for at the inventoried servers. You specify the software scan settings in the Server Inventory policy page. The software scan settings are saved in eDirectory.

By default, the Scanner will not scan for software applications at the inventoried server. You must enable the Software Scan option in the Server Inventory policy. For more information, see [“Configuring the Server Inventory Policy” on page 707](#).

To specify the applications you want to scan for, you add the list of applications or import files that contain the list of applications. You can also export the list of applications as a file and then modify the file.

If you have a large number of software applications that you want to specify, you can create a Custom Scan file following the conventions explained in this section and later import the file.

To specify software scan settings that you specified at a different location, you export the file at that location and import the file at the location you want to use the list.

The following sections contain more information to help you customize scanning of the inventoried servers:

- ◆ [“Adding New Applications for Scanning” on page 831](#)
- ◆ [“Format of the Custom Scan File” on page 832](#)
- ◆ [“Exporting the List of Application Files for Scanning” on page 833](#)

Adding New Applications for Scanning

To add a new application, you must provide the details of the application.

To add a new application for scanning:

- 1 In ConsoleOne, open the Server Inventory policy.

For more information, see [“Configuring the Server Inventory Policy” on page 707](#).

Ensure that the Enable Software Scan option is checked.

- 2** Click the Custom Scan Editor button.
- 3** Click Add to specify the details of the application.
- 4** Fill in the details of the application:
Vendor name, Product name, Product version, Filename, File Size (in Bytes)
- 5** Click OK.
- 6** To save the application entry in eDirectory, click OK in the Custom Scan Editor dialog box.

You can also add application entries to the Custom Scan table by importing a file with the list of application entries. You create this file by following the format of the Custom Scan file conventions. For more information, see [“Format of the Custom Scan File” on page 832](#).

To add a list of new applications:

- 1** Open a text editor.
- 2** Create a file with the format specified in [“Format of the Custom Scan File” on page 832](#).
- 3** Save the application as a text file with any extension you prefer.
- 4** In ConsoleOne, open the Server Inventory policy.
Ensure that the Enable Software Scan option is checked.
- 5** Click Custom Scan Editor.
- 6** Click Import.
To save the application entry in eDirectory, click OK in the Custom Scan Editor dialog box.

Format of the Custom Scan File

The contents of the Custom Scan file are as follows:

total_number_of_application_entries_in_Custom_Scan_file;
total_number_of_columns_in_the_application_entry

vendor_name;product_name;product_version;file_name;file_size (in Bytes)

vendor_name;product_name;product_version;file_name;file_size (in Bytes)

vendor_name;product_name;product_version;file_name;file_size (in Bytes)

Keep in mind the following guidelines as you work with the Custom Scan file:

- ◆ The default total number of columns in the application entry is 5.
- ◆ The separator between the columns is a semicolon (;).
- ◆ Fill in all the columns for each application entry.
- ◆ Do not use comma (,) in the file size parameter.

The following is a sample Custom Scan file:

```
2;5
Novell;GroupWise;5.5;grpwise.exe;4025856
Novell;client32nlm;3.03;client32.nlm;524168
```

Exporting the List of Application Files for Scanning

You can export the Custom Scan file to use at a different location. You export the Custom Scan file at one location and then import it at the other location.

To export the list of applications:

- 1 In ConsoleOne, open the Server Inventory policy.

For more information, see [“Configuring the Server Inventory Policy” on page 707](#).

Ensure that the Enable Software Scan option is checked.

- 2 Click Custom Scan Editor.
- 3 Click Export.
- 4 Type the filename with any extension for the text file.

The export file is a text file.

- 5 Click OK.

The exported file will contain the list of applications that are displayed in the Custom Scan table. If you have not saved the list of applications before exporting, the entries in the exported file and the saved application entries in eDirectory will differ.

Scanning for Vendor-Specific Asset Information from DMI

- 1 In the Server Inventory policy, click the Configuration Editor tab.

For more information, see [“Configuring the Server Inventory Policy” on page 707](#).

- 2 Click the Asset Information suboption, then click Set Defaults.

The following entries will be populated.

```
[ASSETTAG]
DMI1_CLASSNAME=
DMI1_ATTRIBUTEID=
DMI2_CLASSNAME=
DMI2_ATTRIBUTEID=
[SERIALNUMBER]
DMI1_CLASSNAME=
DMI1_ATTRIBUTEID=
DMI2_CLASSNAME=
DMI2_ATTRIBUTEID=
[MODEL]
DMI1_CLASSNAME=
DMI1_ATTRIBUTEID=
DMI2_CLASSNAME=
DMI2_ATTRIBUTEID=
```

```
[COMPUTERTYPE]DMI1_CLASSNAME=DMI1_ATTRIBUTEID=  
[MODELNUMBER]DMI1_CLASSNAME=DMI1_ATTRIBUTEID=
```

3 Specify the values.

The Asset Information contains the following sections:

- ◆ Contains Asset Tag in the section [ASSETTAG]
- ◆ Contains Serial Number in the section [SERIALNUMBER]
- ◆ Contains Computer Model in the section [MODEL]
- ◆ Contains Computer Type [COMPUTERTYPE]
- ◆ Contains Computer Model Number [MODELNUMBER]

Each section contains the particular DMI Class name and DMI Class Attribute ID.

The format of Asset Information is as follows:

```
[ASSETTAG]  
  
DMI1_CLASSNAME=DMI_class_pathname_for_asset_tag  
DMI1_ATTRIBUTEID=DMI_attribute_ID_for_asset_tag  
  
[SERIALNUMBER]  
  
DMI1_CLASSNAME=DMI_class_pathname_for_serial_number  
DMI1_ATTRIBUTEID=DMI_attribute_ID_for_serial_number  
  
[MODEL]  
  
DMI1_CLASSNAME=DMI_class_pathname_for_computer_model  
DMI1_ATTRIBUTEID=DMI_attribute_ID_for_computer_model
```

The value of the Asset Information sections can have a maximum string length of 64 characters.

A DMI Class name can be any DMI class other than DMTF|COMPONENTID|00x.

If there is more than one DMI vendor implementing different custom DMI classes, you can specify multiple DMI classes. A maximum of five classes can be specified in these sections. For example, the asset information for five classes is as follows:

```
[ASSETTAG]  
  
DMI1_CLASSNAME=DMI_class_pathname_for_asset_tag  
DMI1_ATTRIBUTEID=DMI_attribute_ID_for_asset_tag  
DMI2_CLASSNAME=DMI_class_pathname_for_asset_tag  
DMI2_ATTRIBUTEID=DMI_attribute_ID_for_asset_tag  
DMI3_CLASSNAME=DMI_class_pathname_for_asset_tag  
DMI3_ATTRIBUTEID=DMI_attribute_ID_for_asset_tag  
DMI4_CLASSNAME=DMI_class_pathname_for_asset_tag  
DMI4_ATTRIBUTEID=DMI_attribute_ID_for_asset_tag  
DMI5_CLASSNAME=DMI_class_pathname_for_asset_tag
```

```
DMI5_ATTRIBUTEID=DMI_attribute_ID_for_asset_tag
```

The scanner will process DMI1 and if the values of DMI1 are valid, the scanner will not process the remaining DMI classes.

- 4** Click OK.
- 5** Run the scans on the inventoried servers.
Verify that the inventory information is in the Inventory Summary window.

Customizing the Software Scanning Information of Vendors and Products

The software information of the same vendor may sometimes have different vendor names or product names. For example, if the software scan data contains information of more than one product for the same vendor, and if the vendor name differs, the inventory display windows will display the software information under different vendor names.

By default, the software information is displayed for each unique vendor name in the Inventory Query window, Inventory Summary window, and the Inventory reports. If the vendor or product names differ, you can merge the software information. You can also prevent the display of specific vendors and products in the inventory windows. You customize these settings in the Software Rules.

To customize the vendor and product names for display:

- 1** In the Server Inventory policy, click the Configuration Editor tab.
For more information, see [“Configuring the Server Inventory Policy” on page 707](#).
- 2** Click the SWRules suboption, then click Set Defaults.

The default values are displayed.

```
[vendor]
Novell=Novell Incorporated
Novell Inc=Novell Incorporated
Novell Corporation=Novell Incorporated
Novell Corp=Novell Incorporated
Microsoft=Microsoft Corporation
..
[PRODUCT]
Microsoft® Windows Operating System=NULL
Microsoft © Windows(TM) Operating System=NULL
Microsoft(R) Windows NT(R) Operating System=NULL
Microsoft(R) Windows (R) 2000 Operating System=NULL
..
```

- 3** Add or modify the entries.
The format of SWRules is as follows:

```
[vendor]
```

```
scanned_vendor_name_reported_by_scanner= vendor_display_name_you_specify
scanned_vendor_name_reported_by_scanner= vendor_display_name_you_specify
[product]
scanned_product_name_reported_by_scanner= product_display_name_you_specify
scanned_product_name_reported_by_scanner= product_display_name_you_specify
```

You should follow these rules while editing SWRules:

- ◆ Ensure that blank lines do not exist between the sections.
- ◆ The section should end with a carriage return.
- ◆ Ensure that spaces and symbols in the *scanned_vendor_name_reported_by_the_scanner* and *scanned_product_name_reported_by_the_vendor* do not exist. The scanners compare the *scanned_vendor_name_reported_by_the_scanner* and the *scanned_product_name_reported_by_the_scanner* with the scanned data that they collect. Ensure that names that you use are not case sensitive.

If you specify incorrect entries, the entries preceding the incorrect entry will be used and the other entries will be ignored.

- ◆ To modify the vendor name, specify the details for *scanned_vendor_name_reported_by_scanner* and the *vendor_display_name_you_specify*.

For example, to display the software vendor information for Novell, Novell Inc., Novell Corp, and Novell Inc as Novell Inc., edit the following section:

```
[vendor]
Novell=Novell Inc.
NOVELL INC=Novell Inc.
NOVELL CORP=Novell Inc.
NOVELL Inc=Novell Inc.
```

- ◆ To modify the product name, specify the Scanned Product Name and the Product Display Name.

For example, to display the product information: Novell NetWare (TM) Operating System, Novell NetWare[®], Novell NetWare (R) Operating System as Novell NetWare[®], edit the following section.

```
[product]
Novell NetWare (TM) Operating System=Novell NetWare®
Novell NetWare=Novell NetWare®
Novell NetWare (R) Operating System=Novell NetWare®
```

- ◆ To specify that the scanned information for a product or vendor should not be reported by the scanners, add the following entry:

```
[vendor]
others=null
```

4 Click OK.

Customizing the Hardware Scanning Information of Jaz and Zip Drive Vendors

The scan information of the vendors for devices such as backup and floppy devices is usually unavailable on the inventoried server. Also, if the information is available, the vendor information does not usually contain the details. You can customize and update information about the vendors of these devices in Server Inventory policy > Configuration Editor > Zipped Names. The scanners read this data during the hardware scanning process for these devices.

To customize and update the vendor information for display:

- 1 In the Server Inventory policy, click the Configuration Editor tab.

For more information, see [“Configuring the Server Inventory Policy” on page 707](#).

- 2 Click the ZIPPED NAMES suboption, then click Set Defaults.

The default values are displayed.

```
[Identifier]
iomega ZIP 100=Iomega 100MB Backup Device
iomega jaz 1GB=Iomega 1GB Backup Device
IOMEGA ZIP 100 D.13=Iomega Corporation
IOMEGA ZIP 1GB D.13=Iomega Corporation
...
```

- 3 Add or modify the entries.

The format of each entry in the section is as follows:

```
[Identifier]
device_id=vendor_display_name_you_specify
```

where *device_id* is the unique ID generated and updated in the registry by the vendor during the installation of the device on the inventoried server.

For example, the contents of the section are as follows:

```
[Identifier]
iomega ZIP 100=Iomega 100MB Backup Device
```

This entry is for a 100 MB Zip* drive installed on the inventoried server.

If you specify incorrect values for the device ID entry, the device will not be displayed in the Inventory windows.

- 4 Click OK.

Exporting the Inventory Data to CSV Format

You can customize the inventory data you want to export from the ZfS Inventory database in to a comma-separated value (.csv) file.

You select the inventory components that should be exported, such as the Operating System Name and Version. You can further filter the inventoried servers whose attributes will be exported. For example, you can export only those inventoried servers with a particular processor speed. The Data Export tool will export all inventoried servers satisfying these query conditions into a .csv file.

If you want to reuse the same data export settings for export, you can save the data export configurations.

The following sections will help you use the Data Export tool:

- ◆ [“Invoking the Data Export Tool” on page 838](#)
- ◆ [“Exporting the Inventory Data to a CSV File” on page 838](#)
- ◆ [“Forming the Query and Setting the Filter Conditions” on page 839](#)
- ◆ [“Loading an Existing Configuration File” on page 841](#)
- ◆ [“Running the Data Export Program from the Inventory Server” on page 842](#)

Invoking the Data Export Tool

To invoke the Data Export tool:

- 1** In ConsoleOne, select a container.
- 2** Invoke the Data Export tool.
 - ◆ To invoke the Data Export tool from a database object, right-click the database object, click ZENworks Inventory, then click Data Export.
 - ◆ To invoke the Data Export tool from the ConsoleOne Tools menu, you must first configure the Inventory database and then click Tools > ZENworks Inventory > Data Export. For more information on how to configure the Inventory database, see [“Configuring the Inventory Database” on page 812](#).

Exporting the Inventory Data to a CSV File

To export the inventory data to a .CSV file:

- 1** Open the Data Export tool. See [“Invoking the Data Export Tool” on page 838](#).
- 2** Select Create a New Database Query.

This option lets you add a new query that defines the inventory components such as hardware, software, network, and others that you want to export. You can also specify the criteria to limit the inventoried servers to be included in the query. Based on the inventory components and criteria you specify, the inventory data from the database is exported to a .csv file.

Click Next.

- 3** Specify the filter conditions for the inventoried servers.
 - 3a** Click Edit Query. For more information to how to define a query, see [“Forming the Query and Setting the Filter Conditions” on page 839](#).
 - 3b** If you have formed a query with only software attributes (such as Vendor, Name, Version, and Product Identification), the Enable Filter check box will be available for selection.

If you want the results that will be stored in .csv file to be filtered on the basis of the above query, select the Enable Filter check box.
 - 3c** Click Next.

- 4 Select the database fields from the list of Database Fields, then click Add.

If you select a group component, all subcomponents of the group are added. For example, if you select the Software component group, the subcomponents of Software such as vendor name, product name, and version are added.

Click Next.

- 5 View the data export settings.

- 5a Click Save Configuration to save the configurations settings to an .exp file, specify the filename for the .EXP file, then click Save.

The configuration file (.exp) contains the settings such as the inventory components you selected, and also the query formed for filtering the inventoried server data export. You create an .exp file so that you can reload the configuration settings and generate the .CSV files any time you need to.

- 5b Click Next.

- 6 Select Perform the Query from This Computer to run the data export processing from the workstation computer. This option will access the Inventory database on the specified database server and export the data in to a .csv file.

If you want to apply default encoding of the machine to the .csv file, select Default Encoding. The Default Encoding check box is selected by default. To apply Unicode encoding to the .csv file, select Unicode Encoding.

- 7 Specify the .csv filename, then click Finish.

This generates the .CSV file in the specified directory. Open the .csv file in Microsoft Excel or any other CSV-supported viewer to view the exported data.

- 8 To run the data export tool from an Inventory server, select the Perform the Query option on a Remote Inventory server. See [“Running the Data Export Program from the Inventory Server” on page 842](#).

- 9 Save the configuration settings, if necessary.

- 10 Click Finish.

If the configuration settings have not been saved, you will be prompted to save the changes.

Forming the Query and Setting the Filter Conditions

To form the query and set the filter conditions for the data export:

- 1 In ConsoleOne, open the Data Export tool. See [“Invoking the Data Export Tool” on page 838](#).
- 2 Select Create a New Database Query.
- 3 Set the scope for exporting the data from the Inventory database.

If the ConsoleOne snap-ins and the Data Export tool have been installed for both ZENworks for Servers 3 and ZENworks for Desktops 4, the Data Export tool allows you to change the scope of exporting the inventory data.

By default, the Servers option will be enabled. The query locates all inventoried servers satisfying the query expression. If ZENworks for Servers and ZENworks for Desktops are installed in the same environment, the Workstations, the Servers and the Both options will be available. When you select Servers, the query locates all inventoried servers satisfying the query expression. Choose Both to include all inventoried workstations and inventoried servers satisfying the query expression.

Also, you must reconfigure the following Database query conditions:

Selecting the Attributes of the Inventory Components: In the Select Attribute window, click the Browse Attribute button to select component attributes. For example, to specify the version of BIOS as a component in the data export, select BIOS as the component, and select Version as the component attribute.

The components are grouped into the following categories: General, Software, Hardware, Network, and System.

The custom attribute will be prefixed by an asterisk (*).

Machines that do not satisfy the query: Select the check box to retrieve machines that do not satisfy the query. By default, this check box is not selected.

Relational operators: The Relational operators determine the relationship between the components and the value. They are grouped on the basis of data type of the attribute selected in the Select Attribute window as shown in the following table:

Data Type of the Attribute	Relational Operators
String	Equal To (=), Not Equal To (!=), Matches ([]), Does Not Match (![]) and Is NULL (null)
Numeric	Equal (=), Not Equal (!=), Less Than (<), Less Than or Equal To (<=), Greater Than (>), Greater Than or Equal To (>=), and Is NULL (null)
Date	On (=), After (>), On or After (>=), Before (<), On or Before (<=), and Is NULL (null)
Enum	Equal To (=), Not Equal To (!=), and Is NULL (null)
Custom	Includes all the relational operators that are grouped under the String, Numeric, and Date data types

For more information on the usage of the relational operators, see [“Usage of Relational Operators” on page 824](#).

NOTE: If the query does not display the result when the data type of the attribute is Custom and the relational operator is Numeric or Date, use the Equal To operator to find the values for the custom attributes that are stored in the Inventory database.

Values for the inventory attributes: Description values are the possible values of an inventory component. For example, 6.0 is a possible value for the DOS-Version attribute. Description values are not case sensitive.

NOTE: For an enumerated attribute, the value will be displayed in the format, *enumerated_value [enumerated_ID]*. For example, Processor.Processor Family = Pentium (R) III [17].

If you choose Matches ([]) or Does Not Match (![]) as the relational operator, you can use wildcards to substitute characters in the Value field. The following table lists the wildcards that can be used according to the SQL documentation:

Example	Specifies to Include
?	Any one character
_ (underscore)	Any one character
%	Any string of zero or more characters

Example	Specifies to Include
[]	Any one character in the specified range or set
[^]	Any one character not in the specified range or set

NOTE: To define a query using special characters such as ? or [, specify the query in the following formats: [?] or [[]].

The list of description values displayed for an Inventory component is taken from the Inventory database corresponding to the component.

Query connectors and controls: The connectors and controls available for building filter conditions include the following:

AND: The expressions before and after the AND must be true.

OR: Either the expression before the OR or the expression after the OR must be true.

Insert Row: Lets you build the filter condition for this current row.

Delete Row: Deletes the row.

New Group: Lets you form a new filter condition group and specify the criteria for it. This group will be combined with the previous group by using the relational operator specified between the groups.

End: Ends the filter condition.

- 4 Click OK.

Loading an Existing Configuration File

You can load an existing configuration file (.exp). An .exp file contains the settings such as the inventory components you selected, and also the query formed for filtering the inventoried server data export.

After you load the .exp file, you can modify the settings for data export and then export the data to a .csv file.

To load existing configuration settings for data export:

- 1 Ensure that you have generated the data configuration files.

Complete the procedure outlined in [“Exporting the Inventory Data to a CSV File” on page 838](#). This procedure generates the .csv file and the data configuration files.

- 2 In ConsoleOne, open the Data Export tool. See [“Invoking the Data Export Tool” on page 838](#).

- 3 Select Open a Saved Database Query, then click Next.

The default directory for .exp files is `consoleone\consoleone_version\reporting\export`. Click Browse to open an existing .exp file.

If the .exp and .cfg files are invalid or are an older version, the data export will not proceed. The data export displays the number of servers and servers that satisfy the query and filter conditions for export.

- 4 Click a saved database query from the list.

If you want to modify the existing query, click Edit. Otherwise, to proceed with the existing query, click Next.

- 5** View the data export settings. Click Next.
- 6** Select the Perform the Query from this Computer option to run the data export processing from the inventoried server. This option will access the Inventory database on the specified database server and export the data in a .csv file.
- 7** Specify the .CSV filename, then click Finish.

This generates the .CSV file in the specified directory. Open the .csv file in Microsoft Excel or any other CSV-supported viewer to view the exported data.
- 8** To run the data export tool from an Inventory server, click the Perform the Query on a Remote Server option. See [“Running the Data Export Program from the Inventory Server” on page 842.](#)
- 9** Click Finish.

Running the Data Export Program from the Inventory Server

Running the Data Export program from an Inventory server is recommended if you are exporting data from a large database or if you have specified complex queries with more than 20 database fields selected for exporting.

To run the data export program from the Inventory server:

- 1** Ensure that you have generated the data configurations files.

Follow Step 1 to Step 5 outlined in [“Exporting the Inventory Data to a CSV File” on page 838](#) and ensure that you save the settings in the .exp file.

When you save a .exp file, a corresponding data configuration file is created in the same directory with the same filename as the .exp file and with the .cfg file extension.
- 2** Click Perform the Query on a Remote Server to run the data export program from any Inventory server that has Server Inventory components installed, then click Finish.
- 3** Copy the .exp file and .cfg file to the Inventory server.

These two files should exist in the same directory on the Inventory server.

From the Inventory server console, run dbexport.ncf on NetWare[®] servers or dbexport.bat on Windows* NT*/2000 servers, enter

```
DBEXPORT "configuration_filename.EXP" "csv_filename.CSV"
```

where *configuration_filename.exp* is an existing file that contains the data export settings. The data exported from the database will be stored in the *csv_filename.csv*.

In the above command, you must enter the *configuration_filename.exp* and the *csv_filename.csv* filenames within double quotes.

The corresponding .cfg file for the .exp file should be in the same folder as the .exp file. The .cfg file contains the list of the database attributes to be exported.

If the .exp and .cfg files are invalid or are an older version, the data export will not proceed. The data export displays the number of servers and servers that satisfy the query and filter conditions for export.

Open the .CSV file in Microsoft Excel or any other CSV-supported viewer to view the exported data.

30

Monitoring Server Inventory Using Status Logs

Novell® ZENworks® for Servers (ZfS) lets you track whether the scan or the roll-up of information is successful by viewing the log files for scan status, roll-up status and Inventory server status.

The scan status of the inventoried server is reported through local log files.

The inventory components report the status of the Inventory server and roll-up of scan information in Novell eDirectory™.

For example, when you view the status logs, you can determine whether the processing of the scan files was successful or if there were any errors while scanning the server or at the time of roll-up.

You can view the following status information:

- ◆ [“Viewing the Scan Status of an Inventoried Server” on page 843](#)
- ◆ [“Viewing the Roll-Up History of the Inventory Server” on page 844](#)
- ◆ [“Viewing the Status of Inventory Components on an Inventory Server” on page 844](#)
- ◆ [“Viewing the Status of the Last Scan on the Inventoried Server” on page 845](#)
- ◆ [“Viewing the Roll-Up Log for the Inventory Servers” on page 845](#)
- ◆ [“Exporting the Inventory Status Log Files” on page 846](#)
- ◆ [“Overview of Status Logs and Scan Logs” on page 846](#)
- ◆ [“Viewing the Status Log in XML Format” on page 847](#)

Viewing the Scan Status of an Inventoried Server

The Inventory Agent reports status information and errors in the `invagent.log` file. This log file is stored in the `sys:\etc` directory on NetWare® servers and in the `temp` directory or the `windows` directory on Windows* NT* 4.0/2000 servers.

The native scanner reports status information and errors in the `invnative.log` file. This log file is stored in the `sys:\etc` directory on NetWare servers and in the `temp` directory or the `windows` directory on Windows NT 4.0/2000 servers.

The Inventory Policy Enforcer writes the status of the current invocation by the policy engine into the `invagentpolicyenforcer.log` file.

In the `forceDebug=true` mode, the Inventory Agent writes the status of the `.str` file transfer into the `invagentstrtransfer.log` file. This file will be located in the `sys:\system\invscan` directory on NetWare servers and in the `c:\invscan` directory on Windows NT 4.0/2000 servers. (where `C:\` is the root directory if Windows is installed in `c:\winnt`).

Viewing the Roll-Up History of the Inventory Server

The Roll-Up Status reports the status of the roll-up information from the Inventory server that initiated the roll-up of data. For example, if your inventory setup consists of a Leaf Server which initiates the roll-up of data to the next-level Root Server, the Roll-Up log displays the roll-up history of the Leaf Server.

The inventory components of the Inventory server (Sender, Receiver, and Storer) write the scan information in the Roll-Up Status. For example, you view the Roll-Up log to determine whether there were any errors during roll-up of scan data from the Inventory server. This log also displays the most recent roll-up time of the scan data that was stored in the database on the topmost level server (Root Server). This log displays the history of the ten previous roll-up sessions done from the Inventory server.

The following table lists the details of the log:

Status Information	Details
Roll-Up Start Time	Displays the date and time of the roll-up.
Message	Displays the message reported by the inventory component while moving the scan data across the Inventory servers.

You can export the file as a .CSV or tab-delimited file.

To invoke the Roll-Up Status window:

- 1 In ConsoleOne[®], right-click the Inventory Service object, from which the roll-up is done, click Properties, click Status Report tab, then click Roll-Up Status.

Viewing the Status of Inventory Components on an Inventory Server

The Server Status window reports the status of the Inventory server components on the selected Inventory server. You can view the Inventory server Status log for any Inventory Service object. For example, you can determine whether the Sender sent the files to the Receiver or whether the Storer was able to establish the connection with the database successfully. The Server Status window displays the details of the ten latest status messages logged by the Inventory server components.

If the Inventory server components (Sender, Receiver, Selector, Storer, Scan Collector, Service Manager, or Roll-Up Scheduler) are not up and running on the Inventory server, the status of the Inventory server displays the information.

The following table lists the details of the log:

Status Information	Details
Time of Log	Displays the date and time when the message was reported by the inventory components.
Source	Displays the inventory component that has logged the status message.
Message Type	Displays the severity of the message.

Status Information	Details
Message	Displays the message reported by the inventory components.

You can export the log file as a .csv or tab-delimited file.

To view the Server Status window:

- 1 In ConsoleOne, right-click the Inventory Service object, click Properties, click Status Report, then click Server Status.

Viewing the Status of the Last Scan on the Inventoried Server

On NetWare, Windows NT 4.0/2000 servers, the invagent.log and the invnatve.log files will store the details and last execution status of the Inventory scan.

Viewing the Roll-Up Log for the Inventory Servers

The Roll-Up log reports the status of the latest roll-up from the Inventory Service objects in the container. For example, you view the Roll-Up log to determine whether the latest roll-up of information from the Roll-Up server for the Inventory Service object was successful. The inventory components (Sender, Receiver, and Storer) write the roll-up information in the Roll-Up log. You can also choose to display error, warning, and informational status messages of the Intermediate servers.

The following table lists the details of the log:

Status Information	Details
Roll-Up Initiated From	Displays the DN of the Intermediate Server that initiated the roll-up.
Roll-Up Start Time	Displays the date and time the roll-up of information was initiated.
Source	Displays the inventory component that logs the status.
Message Type	Displays the severity of the message.
Message	Displays the message reported by the inventory components while scanning the inventoried server.

You can export the log as a .CSV or tab-delimited file.

To invoke the Roll-Up Log window:

- 1 In ConsoleOne, click the container that contains the Inventory Service object > Tools > ZENworks Inventory > Roll-Up Log.
- 2 Click the severity type of the messages you want to view, then click OK.

Exporting the Inventory Status Log Files

You can store the details of the log files as Comma-Separated-Value reports or as a tab-delimited file.

To save the log as a file:

- 1 In ConsoleOne, open the Status window.
- 2 Click Export, choose the file type, type the filename, then click OK.

Overview of Status Logs and Scan Logs

The following table lists the status logs and scan logs:

Status/Scan Log	Inventory Components that Log the Status	Details of the Log	How to View the Log File
Inventoried Server Scan Log	Scan program, Policy Enforcer	Format module name, time stamp, status code and status message	Available locally on the inventoried server
Roll-Up Log	Sender, Receiver, Storer	Roll-up initiated from, roll-up start time, inventory component, message type, status message	Click the container for the Inventory Service object > Tools > ZENworks Inventory > Roll-Up Log
Invagent.log	Scan program, Inventory Agent	Format module name, time stamp, status code and status message	Opens in any text editor
Invnative.log	Scan program	Format module name, time stamp, status code and status message	Opens in any text editor
Invagentpolicyenforcer.log	Policy Enforcer	Time of log, error type, description, severity and state	Opens in any text editor
Invagentstrtransfer.log (created in the debug mode)	Inventory Agent	Time of log, error type, description, severity and state	Opens in any text editor
Status of Inventory components on Server	Sender, Receiver, Scan Collector, Selector, Storer, Service Manager, Roll-Up Scheduler	Time of log, source, message type, message	In ConsoleOne, right-click the Inventory Service object, click Properties > Status Report > Server Status
Roll-Up Status	Sender, Receiver, Storer	Roll up start time, message	In ConsoleOne, right-click the Inventory Service object, click Properties > Status Report > Roll-Up Status

Viewing the Status Log in XML Format

All inventory components log the status messages in a log file maintained in XML (Extensible Markup Language) format. Unlike the status logs that contain a history of the ten latest status messages, the status XML log stores all status messages.

The log file contains the following data:

- ◆ Inventory module name
- ◆ Date and time of status logging
- ◆ Severity of the message
- ◆ Message text and status message number
- ◆ DN name, if the inventory module is associated with a particular DN object in eDirectory
- ◆ Product-specific details of the module

The format of the log file is as follows:

```
?xml version="1.0" encoding="UTF-8"?>
?xml stylesheet type="text/xsl" href="inventorylog.xsl"?
<message_log>
  <message_entry>
    <module_name>Scanner</module_name>
    <severity>Critical</severity>
    <date_time>8/3/00 12:49 PM</date_time>
    <message_tag>unable to create scan data files
    </message_tag>
    <dn_name>Inv_server</dn_name>
  </message_entry>
  </module_name>Storer</module_name>
  <severity>Critical</severity>
  <date_time>8/3/00 12:49 PM</date_time>
  <message_tag>unable to update the database</message_tag>
  <dn_name>Inv_server</dn_name>
</message_entry>
..
</message_log>
```

A sample style sheet and Document Type Declaration (DTD) file are located in *inventory_installation_directory*\inv\server\xmllog on the Inventory server.

The *inventorylog.xml* log file is located in the *inventory_installation_directory*\inv\server\xmllog directory on NetWare and Windows NT/2000 Inventory servers.

By default, the maximum size of the log file is 100 KB. To modify the maximum size of the log file, edit the `inventorylog.ini` file. On NetWare and Windows NT/2000 Inventory servers, this file is in the `inventory_installation_directory\inv\server\xmllog` directory.

The contents of `INVENTORYLOG.INI` are as follows:

```
max_file_size=100 KB
```

Modify the `MAX_FILE_SIZE` parameter, if required.

If the file size exceeds the value specified in the `MAX_FILE_SIZE` parameter, the file is archived as `filename_old.xml`. The latest messages will be in the current log file.

To view the log data file, use a third-party XML browser.



Documentation Updates

This section contains information on documentation content changes that have been made in the *Administration* guide for Server Inventory since the initial release of Novell® ZENworks® for Servers 3 (ZfS). The information will help you to keep current on changes to the documentation.

If you have purchased ZfS 3.0.2 and have not used or installed ZfS 3 or ZfS 3 SP1, you do not need to review this section.

All changes that are noted in this section were also made in the documentation. The documentation is provided on the Web in two formats: HTML and PDF. The HTML and PDF documentation are both kept up-to-date with the documentation changes listed in this section.

The documentation update information is grouped according to the date the documentation updates were published. Within a dated section, the changes are alphabetically listed by the names of the main table of contents sections for Server Inventory.

If you need to know whether a copy of the PDF documentation you are using is the most recent, the PDF document contains the date it was published on the front title page or in the Legal Notices section immediately following the title page.

The documentation was updated on the following dates:

- ◆ “March 11, 2005” on page 850
- ◆ “November 8, 2004” on page 850
- ◆ “March 10, 2004” on page 851
- ◆ “October 17, 2003” on page 852
- ◆ “June 27, 2003” on page 853
- ◆ “December 18, 2002” on page 854
- ◆ “September 27, 2002” on page 855
- ◆ “June 18, 2002” on page 857
- ◆ “May 17, 2002” on page 858

March 11, 2005

Updates were made to the following section. The changes are explained below.

- ◆ “Setting Up Workstation Inventory” on page 850
- ◆ “Understanding the Server Inventory Components” on page 850

Setting Up Workstation Inventory

The following change was made in this section:

Location	Change
“Setting Up the Inventory Database for Oracle8i and Oracle9i” on page 690	Added a new section, “Setting Up the Inventory Database for Oracle9i” on page 693.

Understanding the Server Inventory Components

The following change was made in this section:

Location	Change
“Understanding the Inventory Removal Service” on page 757	Updated the procedure to remove the inventoried servers from the Inventory database.

November 8, 2004

Throughout the guide, graphics have been updated with font changes and newer icons. No notable content changes were made to any graphics.

March 10, 2004

Updates were made to the following sections:

- ◆ [Setting Up Server Inventory](#)
- ◆ [Understanding the Server Inventory Components](#)

Setting Up Server Inventory

The following changes were made in this section:

Location	Change
“Setting Up the Inventory Database for MS SQL Server 2000” on page 701	Rewrote the section, “Configuring the Inventory Database for MS SQL Server 2000” on page 701 .
“Creating the Inventory Database for Oracle8i on UNIX” on page 691	Specified the Oracle versions (Oracle 8.1.5, 8.1.6, or 8.1.7 Enterprise Edition) supported by the Inventory database on Unix.
“Creating the Inventory Database for Oracle8i on a Windows NT/2000 Server” on page 692	Specified the Oracle versions (Oracle 8.1.5, 8.1.6, or 8.1.7 Enterprise Edition) supported by the Inventory database on a Windows NT/2000 server.
“Configuring the Server Inventory Policy” on page 707	In Step 8f on page 708 , explained how to configure the Custom Scan Editor.
“Stopping the Inventory Service” on page 713	Deleted the following the paragraph: To stop all the Inventory services on a Windows NT/2000 Inventory server, at the server console prompt, execute <code>stopser ""</code> from <code>Inventory_server_installation_directory\in\server\wminv\bin</code> directory.

Understanding the Server Inventory Components

The following changes were made in this section:

Location	Change
“Understanding the Inventory Service Manager” on page 725 > “Services on Windows NT/2000 Inventory Servers” on page 729	<ul style="list-style-type: none">♦ Deleted the following paragraph: To stop all services, enter <code>stopser "*"</code> at the command prompt.♦ Added the procedure to stop the Inventory services on the Windows NT/2000 Inventory server.
“How the Scanners Collect server Inventory Data” on page 731	Updated the information about the Custom Scan Editor.

October 17, 2003

Updates were made to the following sections:

- ♦ [“Setting Up Server Inventory” on page 853](#)
- ♦ [“Managing Inventory Information” on page 853](#)

Setting Up Server Inventory

The following change was made in this section:

Location	Change
“Setting Up the Inventory Database” on page 684	Added a new section, “Setting Up the Inventory Database for MS SQL Server 2000” on page 701 .

Managing Inventory Information

The following changes were made in this section:

Location	Change
“Generating Inventory Reports” on page 827	Added guidelines to be followed while working with the Reporting dialog.
“Running the Data Export Program from the Inventory Server” on page 842	Updated Step 3 on page 842 .

June 27, 2003

Updates were made to the following sections:

- ♦ “Setting Up Server Inventory” on page 853
- ♦ “Managing Inventory Information” on page 853

Setting Up Server Inventory

The following change was made in this section:

Location	Change
“Possible Inventory Server Configurations for a WAN” on page 676	Added a new deployment scenario, “Scenario 7: Deploying Inventory Server Across Firewall” on page 682

Managing Inventory Information

The following change was made in this section:

Location	Change
“Prerequisites for Generating Inventory Reports” on page 824	Added the following prerequisite for generating Inventory reports from an Oracle database: For Oracle, you must install Oracle client only for ODBC because Inventory reports are not compatible with either the older or the later version of the client.

December 18, 2002

Updates were made to the following sections:

- ◆ [“Understanding the Server Inventory Components” on page 854](#)
- ◆ [“Managing Inventory Information” on page 854](#)

Understanding the Server Inventory Components

The following changes were made in this section:

Location	Change
“Understanding the Inventory Scanner” on page 730	<p>Added the following contents in the “Software Information Collected by the Scanners” on page 733 section:</p> <p>If you want to know the mode used by the Scanner to scan the inventoried server, see the inventory summary of the inventoried server > Hardware/Software Inventory > Software > Inventory Scanner Information > Scan Mode.</p> <p>If you want to know the technology available on the inventoried server such as DMI, WMI, and others, see the inventory summary of the inventoried server > Hardware/Software Inventory > General > System Information > Management Technology.</p>

Managing Inventory Information

The following changes were made in this section:

Location	Change
“Running Inventory Reports” on page 824	<p>Added the following note to “Types of Inventory Reports” on page 825 > Software Summary Listing:</p> <p>The Software Summary Listing chart might not be displayed properly because there is too much software data in your Inventory database. For the chart to be displayed properly, use the selection criteria effectively to restrict the results displayed according to your requirement.</p>

September 27, 2002

Updates were made to the following sections:

- ◆ [Understanding Server Inventory](#)
- ◆ [Setting Up Server Inventory](#)
- ◆ [Understanding the Server Inventory Components](#)
- ◆ [Managing Inventory Information](#)

Understanding Server Inventory

The following change was made in this section:

Location	Change
Chapter 25, "Understanding Server Inventory," on page 653	Renamed the chapter, Introduction to Understanding Server Inventory.

Setting Up Server Inventory

The following changes were made in this section:

Location	Change
"Creating the Inventory Database for Oracle8i on UNIX" on page 691	Added the following prerequisite for configuring the Inventory database: To maintain the Inventory database in Oracle, Server Inventory requires that you have a minimum of twenty five Oracle user licenses.
"Configuring the Server Inventory Policy" on page 707	Added the following para in Step 2 on page 707 : Do not select to configure policies in the Solaris or the Linux suboption as they are not supported.
"Starting the Inventory Service" on page 712	Added information on how to start all services on a Windows NT/2000 Inventory server.
"Stopping the Inventory Service" on page 713	Added information on how to stop all services on a Windows NT/2000 Inventory server.

Understanding the Server Inventory Components

The following changes were made in this section:

Location	Change
“Understanding the Inventory Scanner” on page 730	Moved the following sections related to customizing inventory information to “Customizing the Inventory Information” on page 831 in Chapter 29, “Managing Inventory Information,” on page 811 : <ul style="list-style-type: none">◆ “Customizing Software Scanning of Inventoried Servers” on page 831◆ “Scanning for Vendor-Specific Asset Information from DMI” on page 833◆ “Customizing the Software Scanning Information of Vendors and Products” on page 835
“Understanding the Inventory Removal Service” on page 757	Removed the Understanding the Server Removal Service section and added a new section to document how to remove the unwanted, redundant, or obsolete inventoried servers from the Inventory database.
“Understanding the Upgrade Service” on page 759	Added a new section to document about the Upgrade service.
“An Overview of the Inventory Components on the Inventory Server” on page 760	Added Upgrade Service to the Server Inventory component table.

Managing Inventory Information

The following changes were made in this section:

Location	Change
“Managing Inventory Information” on page 811	Renamed the chapter, Viewing the Inventory Information to Understanding Server Inventory .
“Viewing the Inventory Information” on page 812	The following updates were made in this section: <ul style="list-style-type: none">◆ Renamed the section, Displaying Inventory Information to Viewing the Inventory Information.◆ In “Viewing the Inventory Summary of an Inventoried Server” on page 813, updated the table containing the inventory information collected from the inventoried servers.◆ Updated the entire section, “Viewing Inventory Information of Inventoried Servers by Querying the Database” on page 821.◆ Updated the entire section, “Running Inventory Reports” on page 824.

Location	Change
“Customizing the Inventory Information” on page 831	Added a new section documenting how to customize the inventory information.
“Exporting the Inventory Data to CSV Format” on page 837	Updated the entire section.

June 18, 2002

Updates were made to the section, [Setting Up Server Inventory](#).

Setting Up Server Inventory

The following changes were made in this section:

Location	Update
“Organizing the Database Spaces for a Sybase Database on NetWare or Windows NT/2000 Servers (AlterDBSpace Tool)” on page 686	Updated the entire section.
“Understanding the Sybase Database Startup Parameters” on page 688	Corrected the example for the <code>-gc</code> parameter to <code>-gc 120</code> (was incorrectly documented as <code>-gc</code>).
“Backing Up the Inventory Database Running Sybase” on page 689	Corrected the log filename created by the backup tool.
“Creating the Inventory Database for Oracle8i on a Windows NT/2000 Server” on page 692	Increased the number of minimum user licenses required to maintain the Inventory database in Oracle from five to twenty five.
“Manually Creating the Inventory Database Object for Oracle” on page 695	Step 3c: Changed the value of Database (Write Only) Username to <code>MWO_UPDATER</code> (was incorrectly documented as <code>MWO_WRITER</code>).

May 17, 2002

Updates were made to the following sections:

- ◆ [Setting Up Server Inventory](#)
- ◆ [Understanding Server Inventory](#)
- ◆ [Monitoring Server Inventory Using Status Logs](#)

Setting Up Server Inventory

The following change was made in this section:

Location	Change
“Changing the Role of the Inventory Server” on page 713	Updated the entire section.

Understanding Server Inventory

The following change was made in this section:

Location	Change
“Understanding ZfS Inventory Attributes” on page 760	Added descriptions to a few ZfS inventory attributes.

Monitoring Server Inventory Using Status Logs

The following change was made in this section:

Location	Change
Chapter 30, “Monitoring Server Inventory Using Status Logs,” on page 843	Renamed the chapter, Troubleshooting Server Inventory with Status Logs to Monitoring Server Inventory Using Status Logs.

IV

Remote Management

Novell® ZENworks® for Servers (ZfS) Remote Management gives you the ability to manage remote servers from the management console. You can use ZfS to remotely manage NetWare® 4.2/5.x/6 or Windows* NT*/2000 servers.

The Remote Management Agent is installed on each NetWare and Windows server that you want to remotely manage. NetWare 5.1 or 6 provides a Java*-based remote console utility that lets you use a network server manage a remote NetWare server.

Remote Management can save you and your organization time and money. For example, you or your organization's help desk can analyze and remotely fix server problems without having to visit the server, which reduces problem resolution times and increases productivity.

This documentation contains following sections:

- ♦ [Chapter 31, “Remote Management for NetWare Servers,” on page 861](#)
- ♦ [Chapter 32, “Remote Management for Windows NT/2000 Servers,” on page 867](#)
- ♦ [Chapter I, “Documentation Updates,” on page 885](#)

31

Remote Management for NetWare Servers

The Java* -based remote console utility (RConsoleJ) for Novell® ZENWorks for Servers (ZfS) lets you control a NetWare® server and perform the following tasks:

- ♦ Use console commands as you would at the server console
- ♦ Use NLM™ programs as you would at the server console (for example, EDIT.NLM to edit files)
- ♦ Control the server from another server that is using RConsoleJ
- ♦ Upgrade a NetWare server (text-based UI only)

This section contains the following topics:

- ♦ [“Introduction” on page 861](#)
- ♦ [“Deploying” on page 862](#)

Introduction

ZfS RConsoleJ has the following components. These components interact with each other during the remote control session of a NetWare server:

- ♦ [“RConsoleJ Client” on page 861](#)
- ♦ [“RConsoleJ Agent” on page 861](#)
- ♦ [“RConsoleJ Proxy Agent” on page 861](#)

RConsoleJ Client

The RConsoleJ Client is a Java-based utility running on the workstation. From the RConsoleJ Client you can remotely control and monitor all NetWare console operations.

RConsoleJ Agent

The RConsoleJ Agent (RCONAG6.NLM) is a utility running on the target NetWare server. The target NetWare server can be connected over IP, IPX™, or IP/IPX running the RConsoleJ Agent. The RConsoleJ Agent services all RConsoleJ Client requests.

The RConsoleJ Agent advertises its services using the Service Location Protocol (SLP) on a NetWare 5.x and up box.

RConsoleJ Proxy Agent

The RConsoleJ Proxy Agent (RCONPRXY.NLM) is a utility running on a NetWare server (supported only on Netware 5.x and up). It routes all IP packets to IPX and vice versa.

The RConsoleJ Proxy Agent advertises its services using the Service Location Protocol (SLP).

IMPORTANT: If the target NetWare server uses only IPX, the NetWare server (loaded with the RConsoleJ Proxy Agent) must have both IP and IPX stacks installed.

Deploying

Before you install RConsoleJ, ensure that all the installation prerequisites for Remote Control are met.

- ◆ “Setting Up RConsoleJ” on page 862
- ◆ “Initiating RConsoleJ” on page 863
- ◆ “Setting Up Security for RConsoleJ” on page 866

Setting Up RConsoleJ

To set up RConsoleJ, complete the following sections:

- ◆ “Loading the RConsoleJ Agent” on page 862
- ◆ “Running the RConsoleJ Client” on page 862
- ◆ “Loading the RConsoleJ Proxy Agent on a Proxy Server” on page 863

Loading the RConsoleJ Agent

1 At the server console prompt, enter

```
RCONAG6
```

2 Enter the password you want network administrators to use when accessing the target NetWare server using RConsoleJ.

3 Enter the TCP port number.

The default value is 2034.

If the server communicates using IPX only, enter **-1** to disable TCP listening.

To enable listening over a dynamically assigned port, enter **0**.

4 Enter the SPX™ port number on which RCONAG6 will listen for a proxy server.

The default is 16800.

If the server communicates using IP only, enter **-1** to disable SPX listening.

To enable listening over a dynamically assigned port, enter **0**.

NOTE: /DEV/TCP and /DEV/TCPSSL will fail if you are using a pure IPX server.

To enable RConsoleJ across the firewall, you need to keep the following ports open: 2034, 2035, and 2036.

Running the RConsoleJ Client

To run the RConsoleJ Client on a server,

1 From ConsoleOne®, select the target NetWare server object.

2 Click the Tools menu > Remote Management > NetWare.

The Novell RConsoleJ dialog box is displayed.

Loading the RConsoleJ Proxy Agent on a Proxy Server

The NetWare server loaded with RConsoleJ Proxy Agent should have an IP/IPX stack loaded.

- 1 At the server console prompt, enter the following command:

```
RCONPRXY
```

- 2 Enter the TCP port number on which RCONPRXY will listen for RConsoleJ.

The default is 2035.

To enable listening over a dynamically assigned port, enter 0.

When the NetWare server is running the RConsoleJ Proxy Agent, the RConsoleJ Client can communicate through it with the target NetWare server that uses only IPX to communicate.

Initiating RConsoleJ

This section will help you initiate RConsoleJ in the following scenarios:

- ◆ “Scenario 1: An IP Client Controlling an IP NetWare Server” on page 863
- ◆ “Scenario 2: An IP Client Controlling an IPX NetWare Server” on page 864

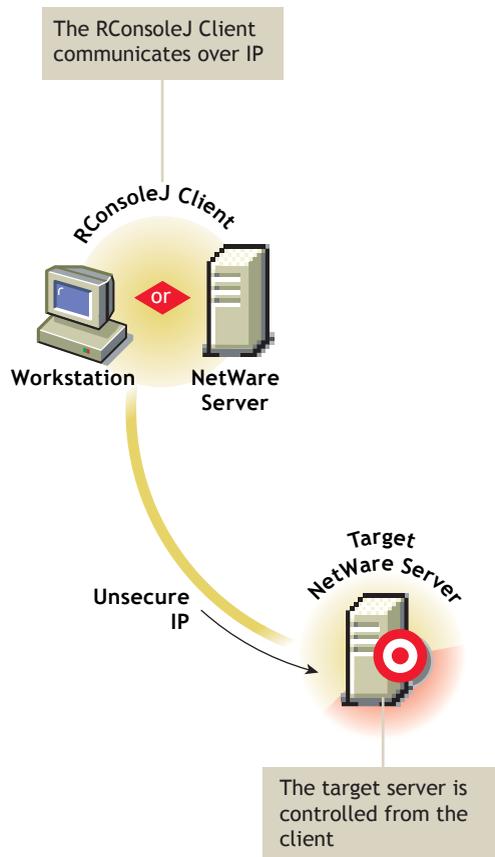
Scenario 1: An IP Client Controlling an IP NetWare Server

Prerequisites

- ◆ “Loading the RConsoleJ Agent” on page 862
- ◆ “Running the RConsoleJ Client” on page 862

Starting an IP Connection

The RConsoleJ Client communicates directly with the RConsoleJ Agent using TCP/IP.



To start an IP connection:

When you run the RConJ client from ConsoleOne, the Novell RConsoleJ dialog box will be displayed with the Netware Server IP address. To run the RConsoleJ client, see [“Running the RConsoleJ Client” on page 862](#).

- 1** Enter the password specified during loading the RConsoleJ Agent.
- 2** Enter the port number.
The default is 2034.
- 3** Click Connect.

Scenario 2: An IP Client Controlling an IPX NetWare Server

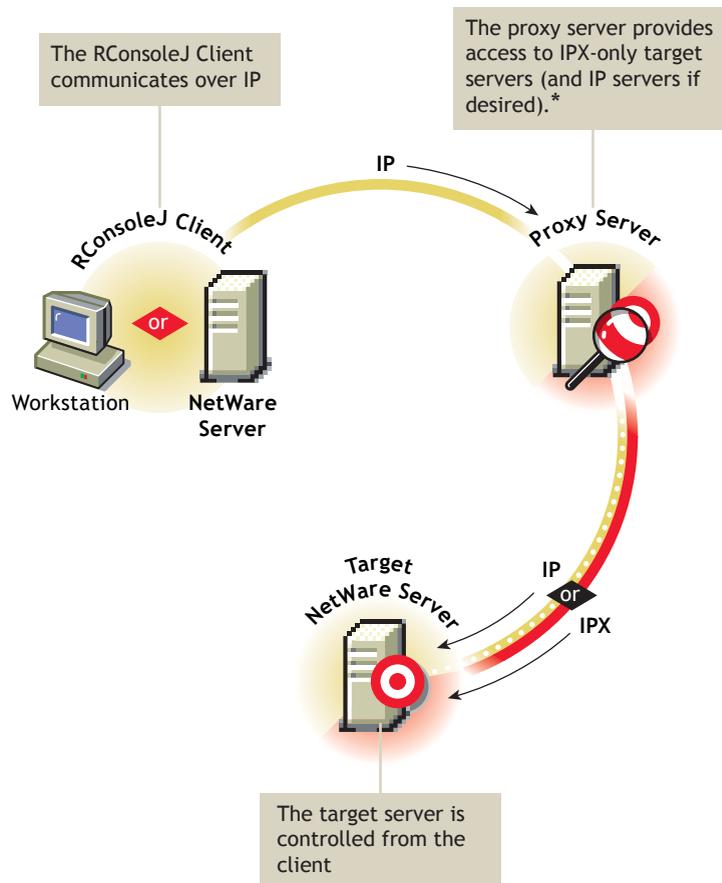
Prerequisites

- ◆ [“Loading the RConsoleJ Agent” on page 862](#)
- ◆ [“Running the RConsoleJ Client” on page 862](#)
- ◆ [“Loading the RConsoleJ Proxy Agent on a Proxy Server” on page 863](#)

Starting an IPX Connection

The RConsoleJ Client communicates with the RConsoleJ Agent through the RConsoleJ Proxy Agent because the target NetWare server is based only on IPX.

The RConsoleJ Proxy Agent is loaded on a NetWare server (proxy server) that has both IP and IPX stacks loaded. The RConsoleJ Proxy Agent receives all the IP requests from the RConsoleJ Client, converts them to IPX requests, and then sends them to the RConsoleJ Agent and vice-versa.



*If the target server uses IPX, the proxy server must have both IP and IPX stacks loaded.

To start an IPX connection:

When you run the RConsoleJ client from ConsoleOne, the Novell RConsoleJ dialog box will be displayed with the Netware Server IP address. To run the RConsoleJ client, see [“Running the RConsoleJ Client” on page 862](#).

- 1** From the Connect type drop-down list, select Connect through Proxy. Select SPX to get the IPX address and select TCP to get the IP address.
- 2** The default port will be selected when you make the above change.
The default is 16800 for IPX address and 2034 for IP address.
- 3** Enter the IP address of the proxy server, or click the Remote Servers icon and then select a proxy server from the list.
- 4** Enter the port number specified during loading the RConsoleJ Proxy Agent.
The RConsoleJ Client communicates with the RConsoleJ Proxy Agent on this port.
The default is 2035.
- 5** Click Connect.

Setting Up Security for RConsoleJ

You can change the agent password to ensure that RConsoleJ sessions are secure.

To change the agent password for a remotely managed NetWare server:

- 1** At the NetWare Console prompt, enter **unload rconag6** to unload RCONAG6.NLM.
- 2** Enter **load rconag6 encrypt**.
- 3** Enter a new password.
- 4** Enter the TCP port number. The default is 2034.
- 5** Enter the SPX port number. The default is 16800.
- 6** Enter **y** when prompted to save the following command line in the LDRCONAG.NCF file.

For Netware 4.x and 5.x:

```
LOAD RCONAG6 - E <encrypted password> <TCP port number> <SPX port number>
```

For Netware 6.x:

```
LOAD RCONAG6 - E <encrypted password> <TCP port number> <SPX port number> <Secure port number>
```

If you enter **n**, the LDRCONAG.NCF file will not be updated. The new password will be valid only for the current session. At a later time, if you load RCONAG6 from the LDRCONAG.NCF file later, the previously saved password will be used.

The new password will be in effect when the agent is loaded from the LDRCONAG script file.

32

Remote Management for Windows NT/2000 Servers

Novell® ZENworks® for Servers (ZfS) Remote Management allows you to remotely view, control, and manage Windows* NT*/2000 servers from your computer.

This chapter contains the following topics:

- ♦ “Remote Management Terminology” on page 867
- ♦ “Understanding Remote Management for Windows NT/2000 Servers” on page 867
- ♦ “Setting Up Security for Remote Management” on page 869
- ♦ “Managing Remote Windows NT/2000 Servers” on page 872

Remote Management Terminology

The following brief glossary provides basic definitions of Remote Management terms:

Managed server: A NetWare® 4.2/5.x/6 or Windows NT/2000 server that you want to remotely view, control, or manage. To remotely view or control a server, you must install the ZfS 3 Remote Management Agent on it. If you want a secure Remote Management session, you must install the ZfS 3 Subscriber component on the managed server.

Management console: A Windows workstation or server running Novell ConsoleOne® with the ZfS 3 Remote Management ConsoleOne snap-ins installed. The management console provides the interface where you manage and administer your network.

Management server: A server with Novell eDirectory™ and the ZfS 3 Distributor components. The eDirectory and Distributor components must be installed only if you want a secure Remote Management session. Your management server can be a managed server or another server.

Remote operator: A user who can remotely view, control, and manage servers.

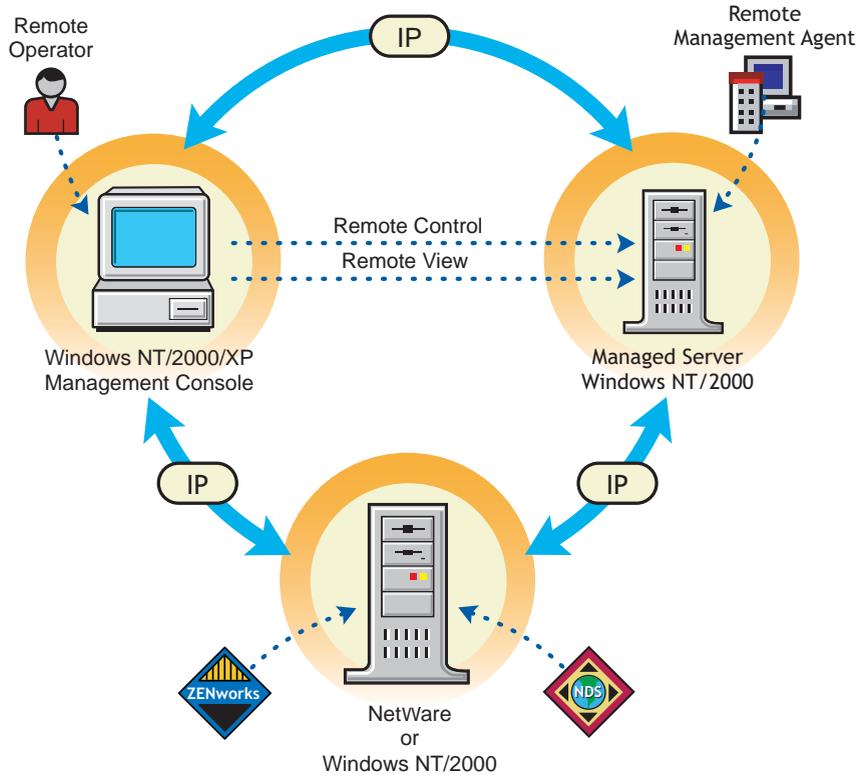
Administrator: A person who has the rights to install Remote Management. All administrators are remote operators but all remote operators are not administrators.

Viewing window: A representation of the managed server desktop. It is displayed on the management console when the remote operator initiates a Remote Management session.

Understanding Remote Management for Windows NT/2000 Servers

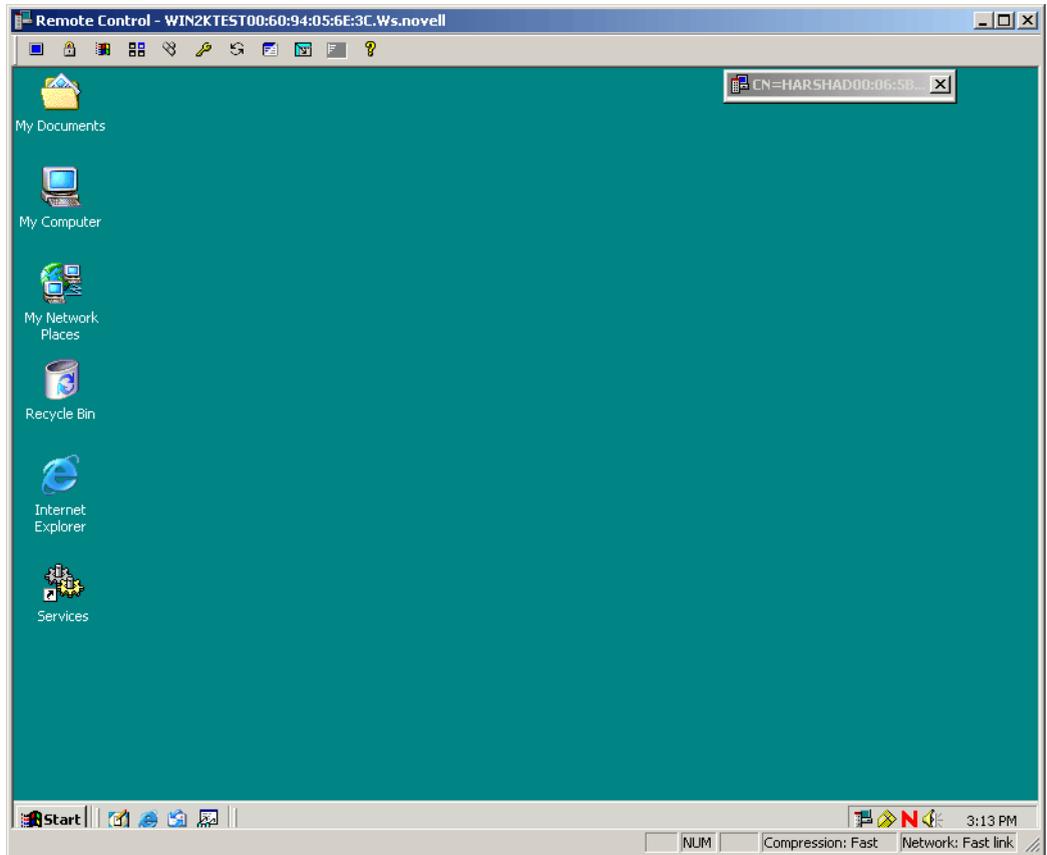
The following illustration depicts the functionality of the ZfS 3 Remote Management, which is explained below:

ZENworks Remote Management



To remotely view or manage a server, the ZfS 3 Remote Management Agent component must be installed on that server. During the ZfS 3 Remote Management installation, the administrator selects the servers and specifies the directory where you want to install the Remote Management Agent. For more information, see [“Installing Remote Management”](#) in the *Installation* guide.

The Remote Management Agent starts automatically when the managed server boots up. When the remote operator initiates a Remote Management session with a managed server, he or she will be prompted to enter the Remote Management Agent password. The agent password can be set either by the administrator during the Remote Management installation or by the user at the managed server after the installation. On successful verification, the Remote Management session proceeds and the Viewing window will be displayed on the management console.



To ensure that the Remote Management session is secure, the user at the managed server can change the agent password. If the user at the managed server sets a new password for the agent, the password set by the administrator during the Remote Management installation will be ineffective. It is very essential that the user at the managed server must communicate the new password to the remote operator each time it is changed. For more details on how to set up a new agent password or change the existing agent password, see [“Setting Up the Agent Password at the Managed Server” on page 872](#).

Setting Up Security for Remote Management

The following sections provide information about setting up security for the Remote Management sessions:

- ◆ [“Configuring the Remote Management Policies” on page 869](#)
- ◆ [“Setting Up the Agent Password at the Managed Server” on page 872](#)

Configuring the Remote Management Policies

To configure the Remote Management policies, you must perform the following tasks:

- ◆ [“Creating the Policy Packages” on page 870](#)
- ◆ [“Creating and Configuring the TED Objects” on page 870](#)
- ◆ [“Configuring the Server Remote Management Policy” on page 871](#)
- ◆ [“Configuring the Distribution Object for Remote Management” on page 871](#)

- ◆ [“Configuring the Distributor and the Subscriber Objects” on page 872](#)

Creating the Policy Packages

ZfS 3 requires policy packages in the eDirectory tree that can hold the server policies. You can later configure and enable the server policies.

Policy packages are eDirectory objects that contain collections of policies grouped according to the object types. You should create an Organizational Unit (OU) for holding the policy packages. Consider the following when determining where to place this OU:

- ◆ Whether you have partitions in your tree
- ◆ The 256-character limit in eDirectory for the full distinguished name
- ◆ How you will use the Search policy to locate the policy package

If you install ZENworks for Desktops (ZfD) to your tree, you may want to keep the ZfS and ZfD policies in separate containers, such as ZfS_Policies and ZfD_Policies.

For ZfS 3 Remote Management, create two containers, one for Tiered Electronic Distribution (TED) objects and the other for the Remote Management policy package.

To create a container:

- 1** In ConsoleOne, right-click the container where you want the container for the policy packages placed.
- 2** Click New > Object > Organizational Unit > OK.
- 3** Name the container, for example, ZfS_Policies > click OK.

IMPORTANT: If you have partitions that are accessed across a WAN, make sure that the Policy Package objects are in the same partition as the Server object so that the Policy/Package Agents will load. Also make sure that the Search policy does not require searching outside the partition where the Server object exists.

For Remote Management, you must create the Distributed Server package. The Distributed Server package is required to distribute the Remote Management policies among the managed servers for enforcement.

To create the Distributed Server package:

- 1** Right-click the policy package’s container > click New > click Policy Package.
The Policy Package Wizard is displayed.
- 2** From the Policy Packages list, select Distributed Server Package > click Next.
- 3** Enter a name for the Distributed Server Package > click Next > click Finish.

Creating and Configuring the TED Objects

For ZfS 3 Remote Management, you must create and configure the following TED objects:

- ◆ TED Distribution
- ◆ TED Channel

To create and configure the TED objects, see [“Tiered Electronic Distribution” on page 371](#).

Configuring the Server Remote Management Policy

The Server Remote Management policy defines the behavior of the Remote Management Agent. This policy is distributed to the specified Windows managed servers using the TED, which helps the remote operator to associate the Remote Management policy to a group of Windows managed servers from the management console.

To configure the Server Remote Management policy:

- 1** In ConsoleOne, right-click the Distribute Server Package object > click Properties.
- 2** Click the Policies tab > select the Windows sub-option.
- 3** Select the check box under the Enabled column for the Server Remote Management Policy.
- 4** Click the Properties button > the Remote Management tab.
- 5** Click the General tab.
- 6** Click Display Remote Management Agent Icon To Users for Remote Control and Remote View sessions.
- 7** Click the Remote Control tab > select the options that you want to use. Your choices are:
 - ◆ Prompt User for Permission to Remote Control
 - ◆ Give User Audible Signal when Remote Controlled
 - ◆ Give User Visible Signal when Remote Controlled
 - ◆ Allow Blanking User's Screen
 - ◆ Allow Locking User's Keyboard and Mouse
- 8** Click the Remote View tab > select the options that you want to use. Your choices are:
 - ◆ Prompt User for Permission to Remote View
 - ◆ Give User Audible Signal when Remote Viewed
 - ◆ Give User Visible Signal when Remote Viewed
- 9** Click Apply > Close.
- 10** Right-click the Server Remote Management policy > select Edit Schedule.
- 11** Modify the schedule > click Apply > click Close.
- 12** To associate the Server Remote Management policy with a managed server, click the Distribution tab.
- 13** Click Add > browse for and select the Distribution object > click OK.
- 14** Click Apply > click Close.

Configuring the Distribution Object for Remote Management

You must configure the Distribution object for distributing the Remote Management policies.

To configure the Distribution object:

- 1** In ConsoleOne, right-click the Distribution object > click Properties.
- 2** Click the Type tab.
- 3** Select Policy Package from the Select Type drop-down list.

- 4** Click Add > select the Distributed Server package that has the Server Remote Management policy.
- 5** Click the Schedule tab.
- 6** Modify the schedule > click Apply > click Close.

Configuring the Distributor and the Subscriber Objects

To configure the Distributor and the Subscriber objects, see [“Tiered Electronic Distribution” on page 371](#).

If the managed servers are residing on a different eDirectory tree or the Windows NT server does not have the eDirectory installed, you must create and configure an External Subscriber object for sending Distributions to Subscribers residing on managed servers in other trees. For more information on External Subscribers, see [“Tiered Electronic Distribution” on page 371](#).

Setting Up the Agent Password at the Managed Server

The user at the managed server can change the password of the Remote Management Agent to make sure that the Remote Management sessions are secure.

To change the agent password:

- 1** Right-click the Remote Management Agent icon from the system tray of the Windows NT/2000 managed server.
- 2** Click Security > click Set Password.
Use a password of ten or fewer alphanumeric characters. The password is case sensitive and cannot be blank.

The new password must be communicated to the remote operator each time it is changed.

Managing Remote Windows NT/2000 Servers

The following sections provide information that will help you effectively manage Remote Management sessions on Windows NT/2000 servers:

- ◆ [“Initiating Remote Management Sessions” on page 872](#)
- ◆ [“Managing a Remote View Session” on page 874](#)
- ◆ [“Managing a Remote Control Session” on page 876](#)
- ◆ [“Viewing the Audit Log for Remote Management Sessions” on page 883](#)
- ◆ [“Improving the Remote Management Performance” on page 883](#)
- ◆ [“Unloading and Reloading the Remote Management Agent” on page 884](#)

Initiating Remote Management Sessions

You have several options for initiating a Remote Management session from ConsoleOne. They include the following:

- ◆ [“Initiating the Remote Management Session from within ConsoleOne” on page 873](#)
- ◆ [“Initiating Remote Management Sessions from the eDirectory/NDS Namespace” on page 873](#)

- ♦ [“Initiating Remote Management Sessions from the Atlas Namespace” on page 874](#)

Initiating the Remote Management Session from within ConsoleOne

- 1** In ConsoleOne, click Tools > Remote Management > Windows.
- 2** In the Remote Management dialog box, enter the IP address or the DNS name of the managed server.
- 3** Enter the agent password.
- 4** Select the Remote Management operation that you want to initiate with the managed server.
- 5** Click OK.

Initiating Remote Management Sessions from the eDirectory/NDS Namespace

You can start a Remote Management session from the eDirectory (NDS) namespace (in ConsoleOne) using one of the following methods:

- 1** In ConsoleOne, click Tools > Remote Management > Windows.
- 2** In the Remote Management dialog box, enter the IP address or the DNS name of the managed server.
- 3** Enter the agent password.
- 4** Select the Remote Management operation that you want to initiate with the managed server.
- 5** Click OK.

You can also use the following procedure:

- 1** In ConsoleOne, select a managed server.
- 2** Click Tools > Remote Management > Windows.
- 3** In the Remote Management dialog box, select the IP address of the managed server from the Agent drop-down list.

The IP address of the selected managed server will be automatically populated to the Agent drop-down list.

- 4** Enter the agent password.
- 5** Select the Remote Management operation that you want to initiate with the managed server.
- 6** Click OK.

You can also use the following procedure:

- 1** In ConsoleOne, right-click a managed server.
- 2** Click Remote Management > Windows.
- 3** In the Remote Management dialog box, select the IP address of the managed server from the Agent drop-down list.

The IP address of the selected managed server will be automatically populated to the Agent drop-down list.

- 4** Enter the agent password.
- 5** Select the Remote Management operation that you want to initiate with the managed server.
- 6** Click OK.

Initiating Remote Management Sessions from the Atlas Namespace

Before initiating a Remote Management session from the Atlas namespace (in ConsoleOne), make sure that the NetWare® Management Agent™ (NMA) is installed and the Discovery discovers the network topology.

To initiate the Remote Management session:

- 1** In ConsoleOne, right-click a managed server.
- 2** Click Actions > click Remote Control or Remote View.
- 3** Select the IP address and enter the agent password.

The IP address of the selected managed server will be automatically populated to the Agent drop-down list.

- 4** Click OK.

Managing a Remote View Session

After you have initiated a Remote Management session and selected Remote View as the operation, you have several options to help you view the managed server.

- ◆ [“Controlling the Display of the Viewing Window” on page 874](#)
- ◆ [“Using the Viewing Window Accelerator Keys” on page 874](#)
- ◆ [“Defining a Custom Accelerator Key Sequence” on page 875](#)
- ◆ [“Stopping a Remote View Session from the Managed Server” on page 876](#)

Controlling the Display of the Viewing Window

You can regulate the display of the Viewing window through using the control options.

To enable the control options:

- 1** Click the Remote Management Agent icon, located at the top left corner of the Viewing window.
- 2** Click Configure.
- 3** To enable the use of accelerator keys on the management console, select Enable Accelerator Keys.
- 4** To use the 16-color palette on the managed server during the Remote Management session, select 16 Color Mode.

Selecting 16 Color Mode will enhance the Remote Management performance.

- 5** To suppress the wallpaper displayed on the managed server’s desktop, select Hide Wallpaper.
- 6** To save the Control Parameter settings, click the Save on Exit check box.

The saved settings will be implemented in the next Remote View session.

- 7** Click OK.

Using the Viewing Window Accelerator Keys

You can use accelerator keys to assign the shortcut keys to the control options and also to control the display of the Viewing window. Default accelerator key sequences are assigned to each accelerator key option. The Accelerator Keys dialog box displays the default key sequence in the

edit field of each accelerator key option. You can define a custom accelerator key sequence to change the default sequence. For more information, see [“Defining a Custom Accelerator Key Sequence” on page 875](#).

To enable the Accelerator Keys option:

- 1 Click the Remote Management Agent icon, located at the top left corner of the Viewing window.
- 2 Click Configure.
- 3 Select Accelerator Keys Enable.
- 4 Click OK.

To open the Accelerator Keys dialog box:

- 1 Click the Remote Management Agent icon, located at the top left corner of the Viewing window.
- 2 Click Accelerator Keys.

The following table explains the Accelerator Key options you can during the Remote View session:

Option	Default Keystroke	Description
Full Screen Toggle	Ctrl+Alt+M	Applicable only if the color resolution settings on the management console and managed server are similar. Sizes the Viewing window to the size of your screen without window borders.
Refresh Screen	Ctrl+Alt+R	Refreshes the Viewing window.
Restart Viewer	Ctrl+Alt+T	Re-establishes the connection with the managed server.
Accelerator Keys Enable	Ctrl+Alt+H	Enables you to change the default accelerator key sequences.
Stop Viewing	Left-Shift+Esc	Closes the Viewing window.
Configure Dialog	Alt+M	Opens the Control Parameters dialog box.
Accelerator Keys Dialog	Alt+A	Opens the Accelerator Keys dialog box.

Defining a Custom Accelerator Key Sequence

The default keystrokes assigned to the accelerator key options are displayed in the edit field to the right of each accelerator key option in the Accelerator Keys dialog box. You can change the accelerator key sequence and define a custom accelerator key sequence if you do not want to use the default keystroke.

To define a custom accelerator key sequence:

- 1 Click the Remote Management Agent icon, located at the top-left corner of the Viewing window.
- 2 Click Accelerator Keys.

- 3** Click the edit field of the accelerator key option where you want to define a custom accelerator key sequence.
- 4** Press the new accelerator key sequence.
- 5** Click OK.

IMPORTANT: The shift keys are left-right sensitive, and are indicated in the Control Options dialog box as Lshift and Rshift.

Stopping a Remote View Session from the Managed Server

You can stop a Remote View session from the managed server using any of the following methods:

- ◆ Right-click the Remote Management Agent icon and click Terminate RC/RV Session.
- ◆ Close the Visible Signal window displayed on the top right corner of the managed server desktop.
- ◆ Press the keystroke that you have defined for the Stop Viewing option in the Accelerator Keys dialog box.

Managing a Remote Control Session

After you have initiated a Remote Management session and selected Remote Control as the operation, you can control the managed server from the management console to provide user assistance and to help resolve server problems. With remote control connections, the remote operator can go beyond viewing the managed server to taking control of it.

You can effectively manage a Remote Control session by performing the following tasks with the Viewing window control options, the Viewing window toolbar buttons, and the Remote Management Agent icon options:

- ◆ [“Controlling the Display of the Viewing Window” on page 876](#)
- ◆ [“Using the Viewing Window Accelerator Keys” on page 877](#)
- ◆ [“Using the Toolbar Buttons on the Viewing Window” on page 878](#)
- ◆ [“Enabling the Wallpaper on the Managed Server” on page 880](#)
- ◆ [“Enhancing the Remote Control Performance Over a Fast Link or a Slow Link” on page 880](#)
- ◆ [“Using the Remote Management Agent Icon” on page 880](#)
- ◆ [“Obtaining Information About Remote Management Sessions” on page 881](#)
- ◆ [“Stopping a Remote Control Session from the Managed Server” on page 882](#)

Controlling the Display of the Viewing Window

You can control the display of the managed server by using the Viewing window control options.

To enable control options:

- 1** Click the Remote Management Agent icon, located at the top left corner of the Viewing window.
- 2** Click Configure.
- 3** Select the control options you want to enable for the remote session.

The following table explains the options you can use to control the display of the Viewing window.

Option	Description
Warn Before Screen Blanking	Informs the user at the management console before the managed server screen is blanked.
Enable Accelerator Keys	Enables the accelerator keys on the management console so that you can change the default accelerator key sequences during the remote session.
16 Color Mode	Forces the use of 16-color palette on the managed server during a Remote Management session. This enhances the Remote Management performance. Use this option only if you are performing the Remote Management session over a slow WAN.
Hide Wallpaper	Suppresses any wallpaper displayed on the managed server. This option is enabled by default. If you want to display the wallpaper on the managed server during a Remote Control or Remote View session, disable this option.
System Key Pass	Passes Alt-key sequences on the management console to the remote Windows NT/2000 server. During a Remote View session, the System Key Pass-Through option is not enabled.
Network Type	If the managed server resides over a LAN, selecting the Fast Links option will accelerate the Remote Management performance. If the managed server is connected over a dial-up link, selecting the Slow Links option will accelerate the Remote Management performance.

- 4 To save the Control Parameter settings, click the Save on Exit check box.

The saved settings will be implemented in the next Remote Control session.

Using the Viewing Window Accelerator Keys

You can use accelerator keys to assign shortcut keys to the control options and also to control the display of the Viewing window. Default accelerator key sequences are assigned to each accelerator key option. The Accelerator Keys dialog box displays the default key sequence in the edit field of each accelerator key option. You can define a custom accelerator key sequence to change the default sequence. For more information, see [“Defining a Custom Accelerator Key Sequence” on page 875](#).

To enable the Accelerator Keys option:

- 1 Click the Remote Management Agent icon, located at the top left corner of the Viewing window.
- 2 Click Configure.
- 3 Select Enable Accelerator Keys.

To open the Accelerator Keys dialog box:

- 1** Click the Remote Management Agent icon, located at the top left corner of the Viewing window.
- 2** Click Accelerator Keys.

The following table explains the Accelerator Key options you can use to control the display of the Viewing window:

Option	Default Keystroke	Description
Full Screen Toggle	Ctrl+Alt+M	Applicable only if the color resolution settings on the management console and managed server are similar. Sizes the Viewing window to the size of your screen without window borders.
Refresh Screen	Ctrl+Alt+R	Refreshes the Viewing window.
Restart Viewer	Ctrl+Alt+T	Re-establishes the connection with the managed server.
Accelerator Keys Enable	Ctrl+Alt+A	Enables you to change the default accelerator key sequences.
Stop Viewing	Left-Shift+Esc	Closes the Viewing window.
Configure Dialog	Alt+M	Opens the Control Parameters dialog box.
Accelerator Keys Dialog	Alt+A	Opens the Accelerator Keys dialog box.
System Key Pass	Ctrl+Alt+S	Passes Alt-key sequences on the management console to the managed server.
Mouse/Keyboard Lock	Ctrl+L	Locks the keyboard and mouse controls at the managed server. This option is available only if the Allow Locking User's Keyboard and Mouse option is enabled in the Server Remote Management policy .
Screen Blank	Ctrl+B	Blanks the screen at the managed server. This option is available only if the Allow Blanking User's Screen option is enabled in the Server Remote Management policy .
Ctrl+Alt+Del	Ctrl+D	Restarts the Windows NT/2000 servers.
Start	Ctrl+S	Opens the taskbar with the Start button on Windows NT/2000 server.
Application Switcher	Ctrl+T	Switches applications on managed servers.

Using the Toolbar Buttons on the Viewing Window

The following table describes the toolbar options in the Viewing window:

Button	Default Keystroke	Key Function
Screen Blanking 	Ctrl+L	<p>Displays only if the Allow Blanking User's Screen option is enabled in the security settings.</p> <p>Blanks the screen at the managed server. When the remote operator selects this option, the screen of the managed server will be blacked out and the operations performed by the remote operator on the managed server will not be visible to the user at the managed server.</p> <p>Not supported over certain display adapters. Refer to the ZfS 3 Readme located at the root of the <i>ZENworks for Servers 3</i> product CD for the list of display adapters that do not support this feature.</p>
Mouse and Keyboard Lock 	Ctrl+B	<p>Locks the keyboard and mouse controls at the managed server. When the remote operator selects this option, the user at the managed server will not be able to use the keyboard and mouse controls of the managed server.</p>
System Start 	Ctrl+S	<p>Sends the Ctrl+Esc keystroke to the managed server.</p> <p>Opens the taskbar with the Start button on Windows NT/2000 servers.</p>
Application Switcher 	Ctrl+T	<p>Sends the Alt-tab key sequences to the managed server.</p> <p>Switches applications on managed servers.</p> <p>To switch the applications,</p> <ol style="list-style-type: none"> 1. In the Viewing window, click the Application Switcher icon or press the Application Switcher shortcut key. 2. To traverse to the application you want using the Application Switcher icon. 3. To view the application, press Tab.
System Key Pass Through 	Ctrl+Alt+S	<p>Sets the system key pass to On or Off.</p> <p>Passes Alt-key sequences on the management console to the managed server.</p> <p>Certain key sequences such as Ctrl+Esc, Alt+Tab, Ctrl+Alt+Del, and Alt+PrintScreen are not allowed even when the System Key Pass-Through is set to On. However, you can use the toolbar buttons on the Viewing window for the Ctrl+Esc, Alt+Tab, and Ctrl+Alt+Del keystrokes.</p>
CTRL+ALT+DEL 	Ctrl+D	<p>Sends the Ctrl+Alt+Del keystroke to the managed server.</p> <p>Displays the Security window on a Windows NT/2000 managed server.</p>

You can define a custom key sequence if you do not want to use the default key sequence. For more information, see [“Defining a Custom Accelerator Key Sequence” on page 875](#).

Enabling the Wallpaper on the Managed Server

When the remote operator initiates a Remote Control session, any wallpaper displayed on the desktop of the managed server will be suppressed. This feature reduces the response time from the managed server for requests from the management console because less traffic is generated over the network while the wallpaper is suppressed.

You can configure the control parameter for this option to change the default settings and enable the display of the wallpaper on the managed server. When you terminate the Remote Control session, the suppressed wallpaper will be restored.

To enable the display of suppressed wallpaper on the managed server:

- 1 Click the Remote Management Agent icon, located at the top left corner > click Configure.
- 2 Deselect the Hide Wallpaper option.

Enhancing the Remote Control Performance Over a Fast Link or a Slow Link

The Remote Control performance, especially over a slow link, has been enhanced through using improved compression.

The performance during a Remote Control session over a slow link or a fast link varies depending on the network traffic. For better response time, try one or more of the following strategies:

- ◆ Deselect the Hide Wallpaper option on the managed server in the Control Parameters dialog box.
- ◆ Assign color settings on the management console higher than the managed server or assign the same color settings for the management console and the managed server.
- ◆ Deselect the Enable Pointer Shadow option before starting the Remote Control or Remote View session.

To deselect Enable Pointer Shadow:

- 1 From the Windows desktop, click Start > Settings > Control Panel > double-click Mouse.
- 2 Click Pointers > deselect Enable Pointer Shadow.

Using the Remote Management Agent Icon

You can manage a remote session from the managed server using the Remote Management Agent icon  options. By default, the Remote Management Agent icon will be displayed in the system tray of the Windows NT/2000 servers. This icon indicates that the Remote Management Agent is loaded on the managed server.

The user at the managed server can right-click the Remote Management Agent icon and choose from the following options:

Option	Description
Terminate Session	Disconnects and closes the remote session on the managed server and displays a message on the management console indicating that the remote session is closed.
Security	Allows the user at the managed server to set or clear the password for the server.

Option	Description
Information	<p>Displays information such as who is accessing the managed server for the remote session, security settings, and the protocol in use for the remote session.</p> <p>For details, see “Obtaining Information About Remote Management Sessions” on page 881.</p> <p>You can right-click or double-click the Remote Management Agent icon to view the Information window.</p>
Help	Displays the Remote Management Agent help file.

Setting Up a Password for the Managed Server

The user at the managed server can set an agent password. This password will override the password set by the administrator during the ZfS Remote Management installation.

To set the agent password:

- 1** From the managed server, right-click the Remote Management Agent icon.
- 2** Click Security > Set Password.

Use a password of ten or fewer alphanumeric characters. The password is case sensitive and cannot be blank.

After the completion of the Remote Management session, you can clear the agent password. If you clear the agent password, the remote operator cannot perform the Remote Management operations.

To clear the agent password:

- 1** On the managed server, right-click the Remote Management Agent icon.
- 2** Click Security > Clear Password.

Obtaining Information About Remote Management Sessions

Using the Information window, the user at the managed server can view details about the session, such as who is accessing the managed server for a remote session, the security settings, and the protocol in use for the remote session.

To view information about remote sessions:

- 1** On the managed server, right-click the Remote Management Agent icon.
- 2** Click Information.
- 3** Click the General tab to view the general information and the Security tab to view the security information.

After you have opened the Information window, you can view different kinds of information about remote sessions on the managed server. See the following sections for details:

- ◆ [“Obtaining General Information” on page 882](#)
- ◆ [“Obtaining Security Information” on page 882](#)

Obtaining General Information

The following table explains the general information you can obtain about Remote Management sessions from the Information window:

Parameter	Description
RM Operation	Lists the ongoing Remote Management sessions.
RM Information > Initiator	Displays the name of the remote operator.
RM Information > Protocol	Displays the protocol that the Remote Management Agent uses to communicate with the management console during a remote session.
RM Information > Optimization	Displays if the optimization driver is enabled or disabled for the Remote Management session. The Remote Management Agent performance will be optimized if the video card on the managed server is compatible with the performance enhancement driver that is installed during Remote Management Agent installation.

Obtaining Security Information

The Security Information dialog box displays information based on the Remote Control and Remote View sessions.

Options	Description
Permission Required	Indicates if the remote operator should obtain permission from the user at the managed server each time the he wants to perform the remote management session on the managed server.
Audible Signal Required	Indicates if an audible signal should be sent to the managed server every time the remote operator accesses the managed server.
Beep Every	Indicates the time interval based on which the audible signal is periodically sent to the managed server.
Visual Signal Required	Indicates if a visible signal should be sent to the managed server every time the remote operator accesses the managed server.
Display Name Every	Indicates the time interval based on which the visual signal is periodically sent to the managed server.
Screen Blanking Allowed	Indicates if the remote operator is allowed to blank the managed server screen. Screen Blanking Allowed is applicable for Remote Control only.
Locking Control Allowed	Indicates if the remote operator is allowed to lock the keyboard and mouse controls of the managed server. Locking Control Allowed is applicable for Remote Control only.

Stopping a Remote Control Session from the Managed Server

You can stop a Remote Control session from the managed server using any of the following methods:

- ◆ Right-click the Remote Management Agent icon and click Terminate RC/RV Session.

- ◆ Close the Visible Signal window displayed on the top right corner of the managed server desktop.
- ◆ Press the keystroke that you have defined for the Stop Viewing option in the Accelerator Keys dialog box.

Viewing the Audit Log for Remote Management Sessions

ZfS records log information on a Windows NT/2000 managed server.

To view the audit log for Remote Management sessions:

- 1** Click Start > Programs > Administrative Tools > Event Viewer.
- 2** Click Log > Application.
- 3** Double-click the event associated with the source Remote Management Agent.

To view only the events pertinent to the Remote Management Agent, choose Remote Management Agent from the source drop-down list in the Filter dialog box.

Improving the Remote Management Performance

The following instructions will help in improving the Remote Management performance:

- ◆ Set the managed server screen resolution to either 640x480 or 600x800.
- ◆ On Windows 2000, deselect the Enable Pointer Shadow option for optimal performance.
To deselect Enable Pointer Shadow:
 - ◆ From the Windows desktop, click Start > Settings > Control Panel > double-click Mouse.
 - ◆ Click Pointers > deselect Enable Pointer Shadow.
- ◆ The speed of the management console depends upon the processing power of the client machine. We recommended that you to use single-processor client with a Pentium* III, 500MHz (or later).
- ◆ At the managed server, use a plain background. Do not set a wallpaper pattern.
- ◆ If the Task manager is opened at the target machine, you can close it or minimize it.
- ◆ Make sure that the scrolling texts (such as the debug windows) and animations are not active on the managed server.
- ◆ Make sure to minimize or close the dialog boxes that are not in use.
- ◆ Use the Page Up and Page Down keys for scrolling through the contents.
- ◆ To perform any operations at the managed server, if possible, use the toolbar options instead of menu options.
- ◆ To maximize remote management visibility, set the management console screen resolution higher than the managed server resolution.
- ◆ If the optimization driver is disabled, set the color settings at managed server to 256 bit.
- ◆ To maximize the Remote Management performance over WAN, do the following at the managed server:
 - ◆ Set the color mode of the managed server to 16 color.
 - ◆ Select the Slow Link Option.

Unloading and Reloading the Remote Management Agent

The following sections explain how you can use the Remote Management Agent during remote sessions:

- ◆ “Unloading the Remote Management Agent” on page 884
- ◆ “Reloading the Remote Management Agent” on page 884

Unloading the Remote Management Agent

You can unload the Remote Management Agent during a remote session. When you unload the Remote Management Agent, the remote session stops. To start another remote session, you will need to reload the Remote Management Agent. For more information, see “Reloading the Remote Management Agent” on page 884.

To unload the Remote Management Agent from a Windows NT managed server:

- 1 From the Control Panel, double-click Services.
- 2 Click Novell ZFS Remote Management Agent > Stop.

To unload the Remote Management Agent from a Windows 2000 managed server:

- 1 From the Control Panel, double-click Administrative Tools.
- 2 Double-click Services.
- 3 Click Novell ZFS Remote Management Agent > Stop.

IMPORTANT: You will be able to stop the Remote Management Agent on Windows NT/2000 only if you have the rights to stop the service.

Reloading the Remote Management Agent

During ZfS installation, the Remote Management Agent is installed on the managed server and started automatically when the managed server starts up. If you shut down the Remote Management Agent during a remote session, the remote session stops. To start another remote session, you need to reload the Remote Management Agent on the managed server.

To reload the Remote Management Agent on Windows NT:

- 1 From the Control Panel, double-click Services.
- 2 Click Novell ZFS Remote Management Agent > Start.

IMPORTANT: You will be able to load the Remote Management Agent on Windows NT only if you have the rights to start the Windows NT service.

To reload the Remote Management Agent on Windows 2000:

- 1 From the Control Panel, double-click Administrative Tools.
- 2 Double-click Services.
- 3 Click Novell ZFS Remote Management Agent > Start.

IMPORTANT: You will be able to load the Remote Management Agent on Windows 2000 only if you have the rights to start the service.



Documentation Updates

This section contains information on documentation content changes that have been made in the *Administration* guide for Remote Management since the initial release of Novell® ZENworks® for Servers (ZfS) 3. This information will help you to keep current on updates to the documentation.

If you have purchased ZfS 3.0.2 and have not used or installed ZfS 3 or ZfS 3 SP1, you do not need to review this section.

All changes that are noted in this section were also made in the documentation. The documentation is provided on the Web in two formats: HTML and PDF. The HTML and PDF documentation are both kept up-to-date with the documentation changes listed in this section.

The documentation update information is grouped according to the date the documentation updates were published. Within a dated section, the changes are alphabetically listed by the names of the main table of contents sections for Remote Management.

If you need to know whether a copy of the PDF documentation you are using is the most recent, the PDF document contains the date it was published on the front title page or in the Legal Notices section immediately following the title page.

The documentation was updated on the following dates:

- ♦ “November 8, 2004” on page 885
- ♦ “May 17, 2002” on page 885

November 8, 2004

Throughout the guide, graphics have been updated with font changes and newer icons. No notable content changes were made to any graphics.

May 17, 2002

Changes were made to the following section:

- ♦ “Remote Management for Windows NT/2000 Servers” on page 885

Remote Management for Windows NT/2000 Servers

The following change was made in this section:

Location	Change
"Remote Management Terminology" on page 867	Defined the term Viewing window.