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* kernel 2.4.*x* (as tested on Red Hat* 7.1 and 7.2)
Solaris* 8

You can administer Policy and Distribution Services using the following:

- **ConsoleOne® 1.3.3,** 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 above is available.

The Policy and Distribution Services documentation contains the following sections:

- Chapter 15, “Configuring Policy and Distribution Services,” on page 423

  After installing ZfS 3 for the first time, use this section to complete a full configuration of your distribution system.

- Chapter 16, “Novell iManager,” on page 481
- Chapter 17, “Tiered Electronic Distribution,” on page 495
- Chapter 18, “Server Policies,” on page 591
- Chapter 19, “Server Software Packages,” on page 641
- Chapter 20, “Desktop Application Distribution,” on page 683
- Chapter 21, “Security in Policy and Distribution Services,” on page 695
- Chapter 22, “Scheduling,” on page 713
- Chapter 23, “Variables,” on page 731
- Chapter 24, “ZENworks Database,” on page 741
- Chapter 25, “Reporting,” on page 771
- Chapter 26, “Documentation Updates,” on page 781

15 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 in 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.

In one planning session, 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 424

  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, you can skip this section.

- "Configuring Your Distribution System" on page 452

  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 469

  This section provides an overview on how you can manage your distribution system using ConsoleOne® and Novell iManager.

- "Configuration Planning Worksheet" on page 470

  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 452, 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 425
2. "Selecting Your Distributions" on page 428
3. "Understanding Your Network Topology" on page 435
4. "Are Additional Distributors Needed?" on page 436
5. "Other Subscribers To Be Installed?" on page 440
6. "Determining the Distribution Flow" on page 442
7. "Understanding Distribution Security" on page 445
8. "Determining the Channels for the Distributions" on page 447
9. "Determining Subscribers' Subscriptions" on page 448
10. "Determining the Distribution Schedules" on page 450

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 425
- "How Is Data Distributed?" on page 426
- "What Will You Need To Know To Plan Your Distribution System?" on page 427

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

Distribution Type	Explanation
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.

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 428.

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 configurable 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 428
- "FTP" on page 430
- "HTTP" on page 430
- "RPM" on page 431
- "Desktop Application" on page 431
- "Policy Package" on page 432
- "Software Package" on page 434

File

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.

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.

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 type keeps a list of the files and directories contained in a Distribution on the source machine. If a source file changes, a new Distribution is built the next time its Build schedule starts and consists of the files that are different between the previous version and the current version. To manually force a Distribution to be built, you can use iManager (see "Forcing TED Agent Actions" on page 490).

For the File type of Distribution, the maximum number of revisions can be set in the Distribution object. When the version number reaches the number you set, the Distributor rebuilds the entire Distribution. 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 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 type of Distribution where their paths and filenames are identical except for case differences.

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.

Determine whether you want to create a File type of 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 type of Distribution

FTP

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.

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 the files and subdirectories are built for the Distribution.

A maximum number of revisions can be set in the Distribution object to conserve disk space. By default, the number is unlimited.

Whenever a Distribution's Build schedule starts, the FTP type will create a new Distribution only if the new version would be different than the previous version.

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 type of 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.

A maximum number of revisions can be set in the Distribution object to conserve disk space. By default, the number is unlimited.

Whenever a Distribution's Build schedule starts, the HTTP type will create a new Distribution only if the new version would be different than the previous version.

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 type of Distribution

RPM

This is a UNIX platform Distribution. You can distribute Red Hat* Package Manager (RPM) packages using the RPM type of Distribution. Any RPM packages you have created can be distributed to your Linux or Solaris servers through TED. However, RPM must first be installed on a Solaris server, because it is not installed with Solaris software by default.

A maximum number of revisions can be set in the Distribution object to conserve disk space. By default, the number is unlimited.

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 type of Distribution

Desktop Application

Distributes ZENworks for Desktops (ZfD) Application objects and associated files to specified locations on the eDirectory tree and target Subscriber servers.

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.

For the Desktop Application type of Distribution, the maximum number of revisions can be set in the Distribution object. When the version number reaches the number you set, the Distributor rebuilds the entire Distribution. By default, this number is 10.

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. For more information, see "Rebuilding Desktop Application Distributions" on page 690.

For more information on integration with ZfD, see Chapter 20, "Desktop Application Distribution," on page 683.

This Distribution type is not supported for Linux and Solaris servers.

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 type of Distribution

Policy Package

In ZfS 3, policies for Subscribers are enforced by being distributed. Previously, they were enforced through context associations.

With the Policy Package type of 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.

Policies for Distributors continue to be enforced through context associations.

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 policies and policy packages, see Chapter 18, "Server Policies," on page 591.

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.
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 600.

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

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.

With this Distribution type 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 if 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 Chapter 19, “Server Software Packages,” on page 641.

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 type of Distribution

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 470. 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 528.

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 437
- “Determining ZfS Software Installation Paths” on page 438
- “Determining Whether a Distributor Server Will Host a ZENworks Database” on page 438
- “Configuring Distributors in a Mixed eDirectory Environment” on page 440

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 533.

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 ZENworks 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 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.

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 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.

For more information, see Chapter 24, "ZENworks Database," on page 741.

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, 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 23, "Variables," on page 731.

  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 533.

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 442
- “Selecting Subscribers for the Distribution Routes” on page 443
- “Configuring the Distribution Routes” on page 444

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 acting as parents for other Subscribers. The hierarchy of Subscribers can be many levels deep. Subscribers that are not used to pass Distributions on to other Subscribers do not need to be in this list.

Subscribers in a Distributor’s routing hierarchy do not need to also receive Distributions from that Distributor. A Subscriber can merely act as a proxy for the Distributor to pass Distributions to other Subscribers.

The Distributor uses its routing hierarchy to determine the route a Distribution will take to get to any given Subscriber.

A 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 Subscribers in the first tier of its routing hierarchy.

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, such as certificates. However, Distributions could require additional security measures that are available in ZfS.

For more information about security, see Chapter 21, "Security in Policy and Distribution Services," on page 695.

Review the following to determine whether you need any additional security for your Distributions:

- "Determining Whether You Need Inter-Server Communications Security" on page 445
- "Determining Whether You Need Encryption Security for Windows Servers" on page 446

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 707.

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 more information on Distribution encryption, see "Distribution Security Using Encryption" on page 702.

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 449
- "Subscriber Groups" on page 449

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

You can create Subscriber Groups for grouping Subscribers that have the same Distribution needs. There is no need to create a Subscriber Group if it will only be associated with one Channel.

Subscriber Groups are useful when you will be sending several different Distributions to the same set of Subscribers. 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 value of a Subscriber Group is when the group is associated with two or more Channels, because 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 a 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 450
- "Determining the Distributors' Refresh Schedule" on page 451
- "Determining the Distribution's Build Schedule" on page 451
- "Determining the Channels' Send Schedules" on page 451
- "Determining the Subscribers' Extract Schedules" on page 452

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 22, "Scheduling," on page 713.

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 22, "Scheduling," on page 713.

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 more information on scheduling issues involving time zones, see "Scheduling Issues" on page 716, especially "Calculating Time Differences" on page 720.

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 453
2. "Setting Up Distributors in a Mixed Network Operating System Environment" on page 456
3. "Setting Up Additional Distribution Security" on page 457
4. "Starting the Distributor Agents" on page 458
5. "Setting Up the Additional Databases" on page 461
6. "Configuring the Distribution Flow" on page 461
7. "Creating the Distributions and Related Channels" on page 464
8. "Subscribing to the Distributions" on page 466
9. "Sending the Distributions" on page 467

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 506 and "Determining How Many Databases You Need" on page 744), 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 453
2. "Starting the Installation Wizard" on page 454
3. "Selecting and Configuring the Distributor and Subscriber Servers" on page 454
4. "Completing the Installation" on page 456

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.

   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 Wizard

1. On the installation workstation, insert the *ZENworks for Servers* product 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 Wizard.

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.

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 Wizard 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 Wizard has finished, review the installation log file to determine whether any components failed to install.

   The log file is located at:

   `C:\TEMP\_RES`*`Number`*`.TXT`

   where *Number* is increased incrementally each time a new installation log is created.

4. If necessary, rerun the Installation Wizard.

   Select only the components that failed to install.

5. Rerun the installation Wizard 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 wizard pages.

Setting Up Distributors in a Mixed Network Operating System Environment

In ZfS 3, 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 457
- "Installing NICI on Windows Servers" on page 458

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 707 for detailed instructions on setting up security for inter-server communications.

Installing NICI on Windows Servers

NICI is used by ZfS for encrypting Distributions. It is installed automatically with NetWare, but must be installed manually for Windows servers.

You only need to install NIC on Windows servers if you will be sending encrypted Distributions to them.

1. On a Windows server (worksheet item 14), insert the *ZENworks for Servers Companion* CD.
2. Run *CD_drive*:\NICI\WIN\WCNICIU0.EXE.
3. Repeat for each applicable Windows server.

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 458
- "Verifying That the Policy and Distributions Services Agents Are Loaded" on page 459

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\smanager\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 440), 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 Distributions Services Agents Are Loaded

To verify that the Policy and Distributions Services agents are running on the target servers:

- “Verifying on NetWare Servers” on page 459
- “Verifying on Windows Servers” on page 460
- “Verifying on Linux or Solaris Servers” on page 460

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 Wizard.

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 Wizard.

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 630.

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 462
- "Configuring Parent Subscribers" on page 463
- "Configuring Subscriber Groups" on page 463

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 fist 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. 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 > 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 Subscribe 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 495. For steps on using the Distribution Wizard to create a File or FTP type of Distribution, see "Using the TED Distribution Wizard" on page 538.

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 464
- "Creating and Configuring the Channel" on page 466

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 "Creating a Distribution" on page 559.

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.

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 1.

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 467
- "Subscribing to the Channels" on page 467

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, select a group of Subscriber objects (worksheet item 23) for servers that you want to use the same extraction schedule > right-click the selected group > click Properties of Multiple Objects.

   Because you can do multiple editing of eDirectory objects, you can select all of the Subscribers that will use the same extraction schedule and edit the Extract Schedule field once for all of them.

2. Click the Schedule tab > click the arrow for the drop-down box > click Run Immediately > click OK.

   This will cause the selected Subscribers to extract the 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 468
2. "Verifying That the Distribution Process Was Successful" on page 468

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 25, "Reporting," on page 771.
- **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 452 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 493.

For information on using ConsoleOne, see the following:

- Chapter 17, “Tiered Electronic Distribution,” on page 495.
- Chapter 18, “Server Policies,” on page 591
- Chapter 19, “Server Software Packages,” on page 641
- Chapter 20, “Desktop Application Distribution,” on page 683
- Chapter 25, “Reporting,” on page 771

For information on using iManager, see “Novell iManager” on page 481.

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 worksheet alone to plan and configure Policy and Distribution Services, even if you feel experienced enough to do so. There are some required steps that are not covered in this worksheet, because planning is not necessary for those steps. Use the sections under "Configuring Your Distribution System" on page 452 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 435.
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 437.

Configuration Information	Instructions
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 440.
4) Database server names:	Enter the server names for each server where you want to install the ZENworks database, which can be installed on NetWare and Windows servers. You can have multiple databases for Policy and Distribution Services, but only one per server. Also enter the purpose for each database, or a Distributor identifier for each database if they will each be used the same way. For more information, see “Determining Whether a Distributor Server Will Host a ZENworks Database” on page 438.

Configuration Information	Instructions
5) Installation paths for Distributors' software:	Enter the path where you want the Distributor software installed. The default is \ZENWORKS for both NetWare and Windows servers. For more information, see "Determining ZfS Software Installation Paths" on page 438.
6) Installation paths for Subscriber software:	Enter the path where you want the Subscriber software installed. The default is \ZENWORKS for both NetWare and Windows servers. For more information, see "Determining ZfS Software Installation Paths" on page 438.
7) Distributors' properties, where different than the installation defaults:	Edit the following information for your Distributor servers: ◆ Distributor object's name (the default is Distributor_*ServerName*) ◆ Distributor's context (using the TED container) ◆ Distributor server's working directory For more information, see "Determining Distributor Properties" on page 437.

Configuration Information	Instructions
8) Subscribers' properties, where different than the installation defaults:	Edit the following information for your Subscriber servers: ◆ Subscriber object's name (the default is Subscriber_*ServerName*) ◆ Subscriber context (using the TED container) ◆ Subscriber server's working directory For more information, see "Other Subscribers To Be Installed?" on page 440.
9) Installation paths for ZENworks database software:	Enter the path where you want the ZFSLOG.DB file located. The default is ZENWORKS\DATABASE. For NetWare servers, we recommend not using the SYS: 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. For more information, see "Determining ZfS Software Installation Paths" on page 438.
10) Database object name:	Either accept the default names, or provide ones that will help you to identify the databases' purposes. For more information, see "Determining Whether a Distributor Server Will Host a ZENworks Database" on page 438.

Configuration Information	Instructions
11) Database object Container:	We recommend you use the same container where your other TED objects reside. For more information, see “Determining Whether a Distributor Server Will Host a ZENworks Database” on page 438.
Configuring the Distributors for a Mixed eDirectory Environment	If you do not have a mixed eDirectory environment, skip to worksheet item 13.
12) IP address of server in eDirectory 8.*x*:	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*. For more information, see “Configuring Distributors in a Mixed eDirectory Environment” on page 440.
Installing Inter-Server Communications	If you do not need to set up inter-server communications, skip to worksheet item 14.
13) Subscriber servers outside your secured network:	Inter-server communications security might be needed if your Distributor and Subscriber servers communicate with servers outside your secured network. For more information, see “Determining Whether You Need Inter-Server Communications Security” on page 445.

Configuration Information	Instructions
Installing NICI on Windows Servers	If you do not need to install NICI to Windows servers, skip to worksheet item 15.
14) Windows, Linux, or Solaris servers (Distributor or Subscriber) that will be involved with Distribution encryption:	List the Windows, Linux, or Solaris servers that will either build (Distributors) or extract (Subscribers) encrypted Distributions. For more information, see “Determining Whether You Need Encryption Security for Windows Servers” on page 446.
Configuring the Distributor Routing Hierarchies	
15) Distributors’ routing hierarchies of tiered Subscribers:	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. For more information, see “Determining the Distribution Flow” on page 442.

Configuration Information	Instructions
Configuring Parent Subscribers	
16) Subscriber/parent Subscriber assignments (end-node Subscribers associated with a parent Subscriber):	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. For more information, see “Selecting Subscribers for the Distribution Routes” on page 443.
Creating and Configuring Subscriber Groups	If you are not using Subscriber Groups, skip to worksheet item 19.
17) Subscriber Group object name:	Enter a unique name for the Subscriber Group. For more information, see “Subscriber Groups” on page 449.
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.

Configuration Information	Instructions
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: ◆ “File” on page 428 ◆ “FTP” on page 430 ◆ “HTTP” on page 430 ◆ “RPM” on page 431 ◆ “Desktop Application” on page 431 ◆ “Policy Package” on page 432 ◆ “Software Package” on page 434 For more information, see “Selecting Your Distributions” on page 428.

Configuration Information	Instructions
Creating and Configuring the Channels	
20) eDirectory container for TED objects:	Container for creating and managing TED objects. 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. For more information, see “Determining Whether a Distributor Server Will Host a ZENworks Database” on page 438.
21) Channel names:	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. For more information, see “Determining the Channels for the Distributions” on page 447.
22) Distributions for the Channels:	Create a list of which Distributions belong to which Channels. For more information, see “Determining the Channels for the Distributions” on page 447.

Configuration Information	Instructions
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 448.
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.

16 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 above 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 482
- "Managing Tiered Electronic Distribution" on page 483
- "Monitoring the Distribution Process" on page 486
- "Monitoring Specific Agents" on page 487
- "Comparing the ZfS Management Role in iManager with ConsoleOne Capabilities" on page 493

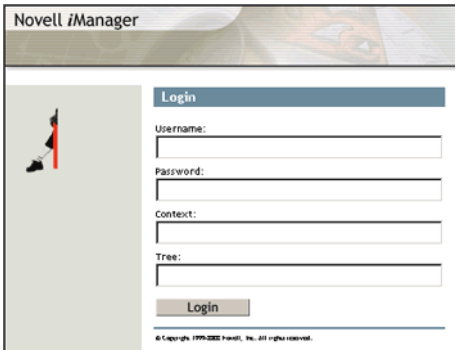
Accessing the ZfS Management Role in iManager

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.

   HINT: 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 483
- "Monitoring the Distribution Process" on page 486
- "Monitoring Specific Agents" on page 487

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
- Subscriber
- Distribution
- Channel
- 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 484
- "Editing TED Object Properties in iManager" on page 485
- "Deleting TED Objects in iManager" on page 485

The following Policy and Distribution Services management tasks cannot be performed in iManager and must be performed using ConsoleOne:

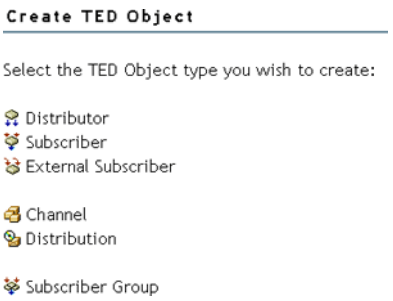
- Creating, editing, and deleting policy packages. See Chapter 18, "Server Policies," on page 591.

- Creating, editing, and deleting software packages. See Chapter 19, “Server Software Packages,” on page 641.
- Creating, editing, and deleting desktop applications. See Chapter 20, “Desktop Application Distribution,” on page 683.
- Managing the Policy/Distribution database. See Chapter 24, “ZENworks Database,” on page 741
- Generating reports from the Policy/Distribution database. See Chapter 25, “Reporting,” on page 771

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.

   You can click Help for assistance with providing the information.
4. Click OK.
5. Continue with “Editing TED Object Properties in iManager” on page 485 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, as described in "Configuring TED Objects" on page 540.

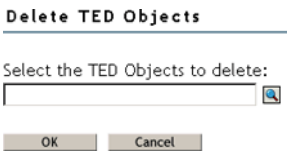
   You can click Help on each property page for assistance with setting the options.

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 488.

Icon	Meaning
	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 488.
	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 488.
	The Subscriber is running ZfS 2 software, rather than ZfS 3 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 487.

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, as described in “Configuring TED Objects” on page 540.

Monitoring Specific Agents

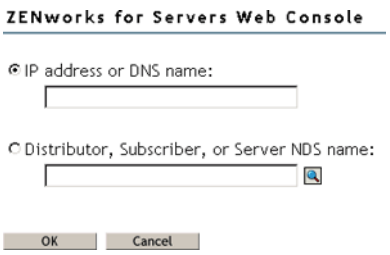
On NetWare® servers, you can monitor the TED agents (Distributor Agent and Subscriber Agent) and the Policy/Package Agent at the server console where they are running. In addition, you can monitor the agents running on any platform (NetWare, Windows, or UNIX*) from Internet Explorer using the ZfS Management role in iManager.

- “Managing the TED Agents from the Remote Web Console” on page 488
- “Managing the Policy/Package Agent from the Remote Web Console” on page 491
- “Opening Multiple Remote Web Console Windows” on page 493

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 assistance in using the features available on that page.

5. Continue with the task that you want to perform:

   - “Managing TED Objects” on page 489
   - “Monitoring TED Agent Status” on page 489
   - “Monitoring Distribution Status” on page 490
   - “Forcing TED Agent Actions” on page 490

- "Managing Security Certificates" on page 491

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	In CSOne?
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	In CSOne?
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

TED Agent (Distributor and Subscriber) Management Task	Remote Web Console Tab and Option	In CSOne?
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	In CSOne?
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	In CSOne?
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	Not with one click
Immediately send to Subscribers all Distributions listed in a selected Channel	Channels > Distribute Channel	Not with one click

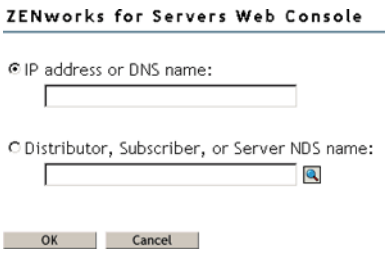
TED Agent (Distributor and Subscriber) Management Task	Remote Web Console Tab and Option	In CSOne?
List the security certificates that are available on a Subscriber	Security > Show Certificates	No
Delete security certificates from a Subscriber	Security > Show Certificates > Delete	No
Have the Distributor 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.



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 assistance in 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

Task	ZfS Management Role in iManager	ConsoleOne
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
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 Resolving certificates Signing CSRs	 Yes No Yes	 No Yes Yes
Managing the Policy/Package Agent	Yes	No

17 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, including the ZfS Management role in Novell iManager.

The following sections provide information for TED:

- "Understanding Tiered Electronic Distribution" on page 496
- "Using the TED Distribution Wizard" on page 538
- "Configuring TED Objects" on page 540
- "Creating Security Certificates" on page 574
- "Scheduling and Sending Distributions" on page 576
- "Refreshing the Distributor" on page 580
- "Forcing a Single Distribution To Be Sent" on page 580
- "Forcing a Channel To Be Sent" on page 581
- "Sending Distributions Through Parent Subscribers" on page 581
- "Sending Distributions Between Trees" on page 582
- "Minimizing Messaging Traffic" on page 583
- "Manually Importing/Exporting Distributions" on page 585
- "Editing the TEDNODE.PROPERTIES File" on page 586

Understanding Tiered Electronic Distribution

Review the following sections for an understanding of Tiered Electronic Distribution (TED):

- "TED Basics" on page 496
- "The TED Architecture" on page 498
- "Relationships of the TED Objects" on page 505
- "Physical Network Connections" on page 506
- "Understanding Distributors" on page 506
- "Understanding Distributor Routing Hierarchies" on page 516
- "TED Issues" on page 526
- "Working Directories" on page 533

TED Basics

The following sections provide basic information on TED:

- "Distribution Management through Tiered Electronic Distribution" on page 496
- "Basic Distribution Concepts" on page 497
- "Key Components of TED" on page 497

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 policy packages.

When you install Policy and Distribution Services, the installation process creates Novell eDirectory™ objects in the 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*, or 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

packages and hosting them in distribution channels for dissemination to the appropriate servers.

TED lets you schedule distributions 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.

Basic Distribution Concepts

The TED distribution concept is as follows:

1. Distributors create certificates to provide Distribution security.
2. Distributions are built on Distributor servers.
3. Distributions are associated with Channels.
4. A Subscriber subscribes to Channels (for all Distributions in those Channels).
5. Certificates are copied to Subscribers for Distribution security verification.
6. The Channel's Distributions are sent from the Distributor to the Subscriber.

Key Components of TED

The key components of TED include:

- eDirectory schema extensions that include objects for Distributors, Distributions, Channels, Subscribers, and External Subscribers
- ConsoleOne® snap-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 and pass them 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

The TED Architecture

To understand how TED can distribute electronic data between servers, you must understand its eDirectory objects, its agents, and the types of distributions it uses:

- “eDirectory Objects” on page 498
- “TED Agents” on page 503

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.

The following eDirectory objects will be extended in the schema of your tree when you install ZfS 3:

- “Distributor” on page 499
- “Distribution” on page 500
- “Channel” on page 501
- “Subscriber” on page 502
- “External Subscriber” on page 503

Distributor

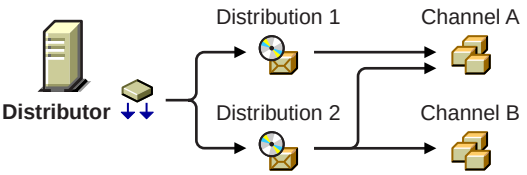
The Distributor object (TED Distributor) is an eDirectory object that defines the properties for the Distributor. You can have multiple Distributor objects in the tree; however, you can only have one Distributor per server.

The Distributor Agent builds a Distribution 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 also 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 for each Distributor, so the digests might not always be available to the Subscribers for a checksum comparison.

A Distributor lists its Distributions in Channels. The Distributor can list any one of its Distributions in several Channels, and several of its Distributions in one Channel. Distributors do not own Channels. However, a Distributor is the sole owner of its Distributions.

The following illustrates a Distributor's relationship with its Distributions and the Channels:



This illustration shows one Distribution being listed in two different Channels.

A Distributor's Refresh schedule determines when it will read eDirectory for changes to its Distributions and other TED objects. A Distributor builds new Distributions it finds and rebuilds any of its Distributions that have changed. The new or rebuilt Distributions are then available to be sent through a Channel according to the Channel's send schedule.

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 three minutes.

A Distributor can build its Distributions any time its Refresh schedule starts, or when it is forced to do so from a server's command line. For more information on scheduling, see Chapter 22, "Scheduling," on page 713.

The Distributor sends its Distributions to Subscribers (generally parent Subscribers. If a Subscriber does not respond to a Distributor (or a parent Subscriber) that is trying to send a Distribution, 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 schedule starts again.

If you delete a Distribution, you should also refresh the Distributor immediately so that it will not try to build a Distribution that no longer exists.

The Distributor contains a distribution route, which is a hierarchical list of Subscribers that indicate the routes the Distributor can take to send a Distribution to a Subscriber.

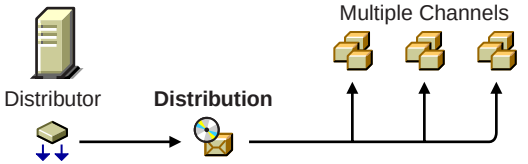
All Distributors can log information to one common database file, or each Distributor can write to its own database file. For more information on databases, see Chapter 24, “ZENworks Database,” on page 741.

Distribution

The Distribution (TED Distribution) object contains a list of data packages or data grouping information. 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.

The following illustrates a Distribution’s relationship with its Distributor and the 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
- Policy Package
- Software Package

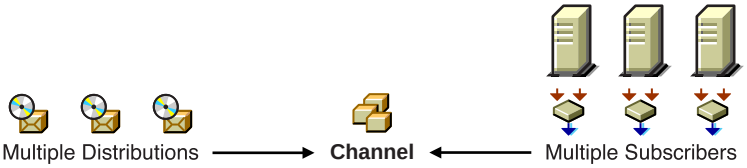
The Desktop Application type is only available when ZENworks for Desktops (ZfD) is also installed.

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 more information on scheduling, see Chapter 22, "Scheduling," on page 713.

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, unless synchronization is enabled, the files and folders will not be removed from the Subscriber server's file system.

Channel

The Channel object (TED Channel) contains a list of Distributions associated with it and Subscribers subscribed to it. The following illustrates a Channel's relationship with Distributions and Subscribers:



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.

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, Subscribers will receive all of the Distributions listed in that Channel.

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 720.

Subscriber

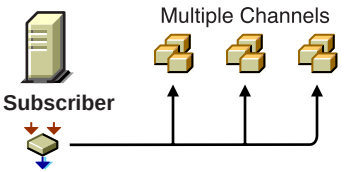
The Subscriber object (TED Subscriber) is an eDirectory object that defines the properties for the Subscriber. The Subscriber is a service that receives and extracts Distributions to obtain the software, files, or policies it needs. Any server where you want to distribute applications, files, or policy packages must be a Subscriber.

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.

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

The following illustrates a Subscriber’s relationship with the Channels:



A Subscriber’s Extract schedule determines when it is available for extracting Distributions. For more information on scheduling, see Chapter 22, “Scheduling,” on page 713.

Subscribers can be parent Subscribers, which act as proxies for the Distributor to pass Distributions on to other Subscribers. This helps the Distributor by providing load-balancing in sending the Distributions to other Subscribers.

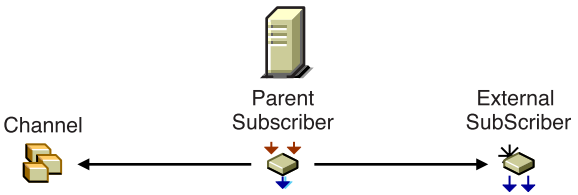
The Subscriber object’s properties lists the parent Subscriber through which it receives all of its Distributions. A Subscriber can receive its Distributions directly from the Distributor if it does not have a parent Subscriber and is not listed in the Distributor’s routing hierarchy.

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.

External Subscriber

The External Subscriber object (TED External Subscriber) is an eDirectory object that represents a Subscriber object in another tree. This allows TED to send Distributions between trees.

The following illustrates an External Subscriber's relationship with its parent Subscriber and the Channel:



External Subscribers are identified by the TCP/IP address or DNS name of the TED Subscriber server in the other tree that it is associated with.

The External Subscriber object's properties lists the Channels it can receive Distributions from. An External Subscriber cannot be a parent Subscriber itself, though it must have a parent Subscriber in the tree where its object resides.

For an External Subscriber that does not have a TED Subscriber object in the eDirectory tree, a TEDNODE.PROPERTIES file can be used. A sample of the TEDNODE.PROPERTIES file is located at the root of the *ZENworks for Servers Companion* CD. You can use this sample file to create a copy to place in the ZENWORKS\PDS\TED directory on the External Subscriber's server. For more information, see "Editing the TEDNODE.PROPERTIES File" on page 586.

For more information, see "Sending Tree-to-Tree Distributions" on page 509.

TED Agents

The following TED agents are used to perform the distribution functions:

- "Distributor Agent" on page 504
- "Subscriber Agent" on page 504

Distributor Agent

The Distributor Agent does the following:

- 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.

This agent is installed on each server where you select to install the Distributor option.

Subscriber Agent

The Subscriber does the following:

- Subscribes to Channels for receiving Distributions.
- Receives and extracts (installs) the following Distribution types to its own file system:

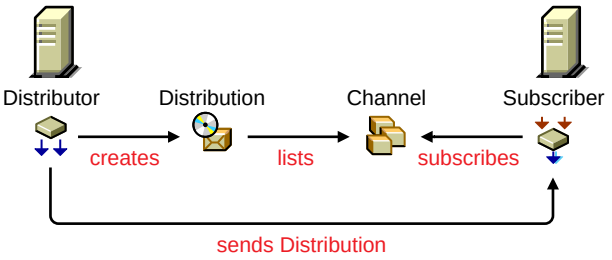
  File
  FTP
  HTTP
  RPM
  Desktop Application

  The Desktop Application type must have ZENworks for Desktops (ZfD) to process it.
- Receives and hands off some Distribution types to the Policy/Package Agent (see "Policy/Package Agent" on page 598).
- As a parent Subscriber, it receives and forwards any Distribution on to other Subscriber servers (even if it was not intended to receive that Distribution itself).

This agent is installed on each server where you select to install the Policy Engine/Subscriber option.

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 or parent Subscriber

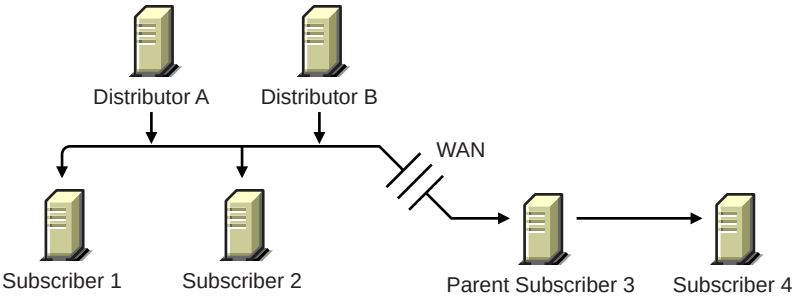
The schedules that need to be coordinated for sending Distributions are the Distributor's Refresh schedule, the Distribution's Build schedule, and the Channel's Send schedule.

The schedules that need to be coordinated for receiving and extracting Distributions are the Channel's Send schedule and the Subscriber's Extract schedule.

For more information on scheduling, see "TED Object Scheduling Issues" on page 720.

Physical Network Connections

The following is an example of the physical network connections between Distributor and Subscriber servers in a WAN environment.



Note the following from the illustration:

- A Subscriber server can be in the same geographic location as the Distributor server
- A Subscriber server can be across the WAN from the Distributor server
- A Distributor server can service multiple Subscriber servers
- A Subscriber server can be serviced by multiple Distributor servers
- A Subscriber server can receive Distributions directly from a Distributor server
- A Subscriber server can receive Distributions indirectly through a parent Subscriber server

Understanding Distributors

Now that you have determined 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 these Distributions, you can determine how many Distributors you will need.

Review the following sections to determine your Distributors:

- "Balancing Workloads" on page 507
- "Distributing Across WAN Links" on page 508
- "Sending Tree-to-Tree Distributions" on page 509

Balancing Workloads

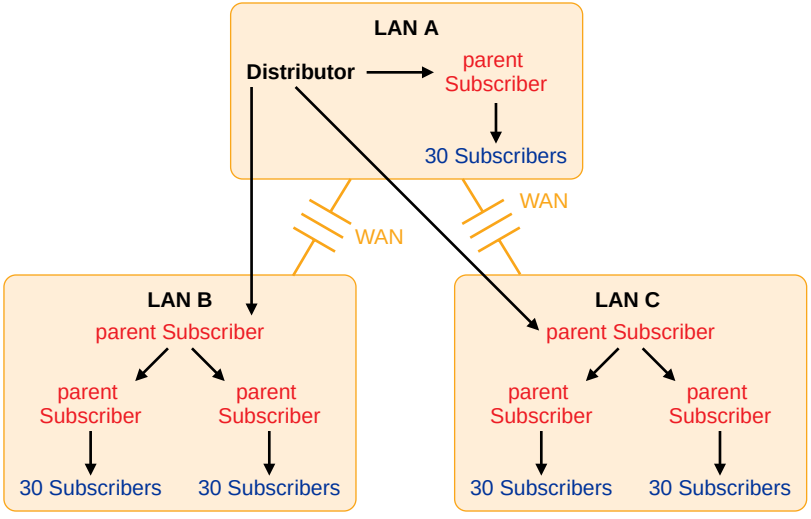
A Distributor can use parent Subscribers in a routing hierarchy to explicitly determine routes for its Distributions. This eases its workload in distributing to Subscribers.

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.

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.

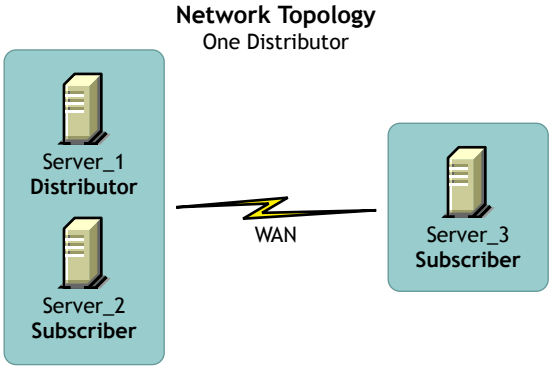
- **Primary Parent Subscribers on a LAN:** You can further minimize WAN traffic by tiering parent Subscribers on the other side of a WAN link from the Distributor. In other words, you can have just one parent Subscriber in the routing hierarchy that would also be a parent to several other parent Subscribers 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.

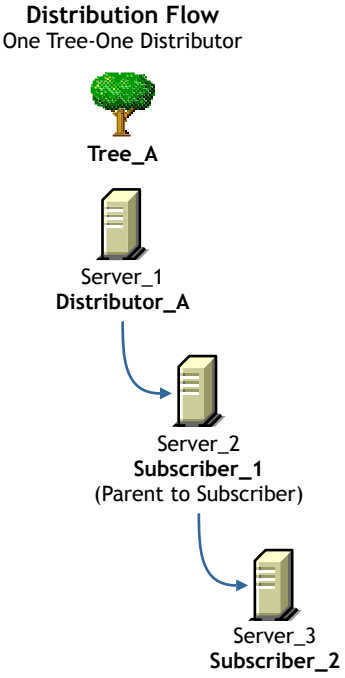
Sending Tree-to-Tree Distributions

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).

Review the following sections to understand using External Subscribers for tree-to-tree distributions:

- "Using External Subscribers" on page 511
- "External Subscribers, One Distributor, and Multiple Trees" on page 512
- "External Subscribers, Multiple Distributors, and Multiple Trees" on page 513
- "Understanding External Subscribers" on page 514

Using External Subscribers

If you installed all of your TED objects on one tree, an External Subscriber is not necessary, because you can send your Distributions using the objects in the one tree. However, if you installed ZfS objects in several different trees, then you can use External Subscribers for sending Distributions between these trees.

TED uses IP addresses as a pointer to a NetWare or Windows server. Therefore, Subscriber objects do not need to be in the same tree as their NetWare servers' eDirectory objects.

The properties of an External Subscriber object contains the following:

- IP address of a Subscriber server
- Context of a parent Subscriber in the External Subscriber's own tree (optional)

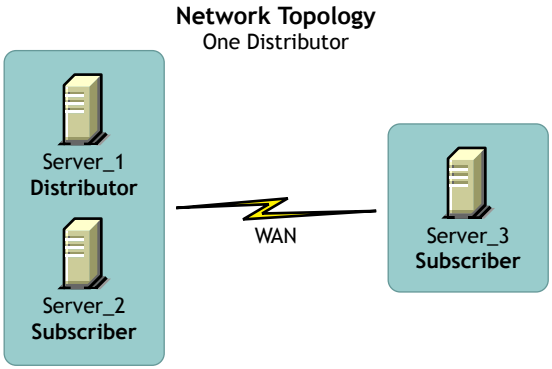
  The parent Subscriber will do the physical work in sending the Distribution file to the server in the other tree.

This allows you to use the External Subscriber for sending a Distribution to a Subscriber in another tree.

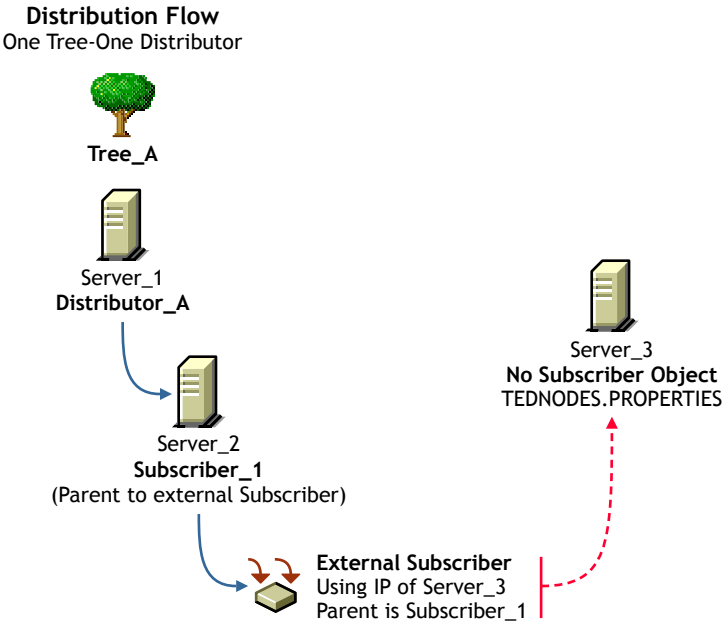
External Subscribers, One Distributor, and Multiple Trees

Once you have installed Policy and Distribution Services software to servers in multiple trees, you can send Distributions to those servers using the Subscriber and External Subscriber objects.

If you network topology were:



You could have the following TED configuration for the Distributor's routing hierarchy:



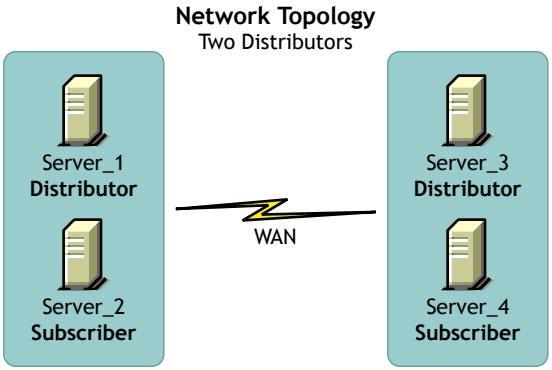
In this example, the Server_3 does not have a Subscriber object in either tree. Therefore, an External Subscriber object is created in Tree_A so that Server_3 can receive Distributions from Tree_A. However, Server_3 must have the Subscriber software installed on it in order to receive and extract the Distribution.

Because there is no Subscriber object for Server_3 in Tree_A, Server_3 will not receive any configuration information from the Distributor in Tree_A. Therefore, it must have the TEDNODES.PROPERTIES file installed on the server where configuration information is kept.

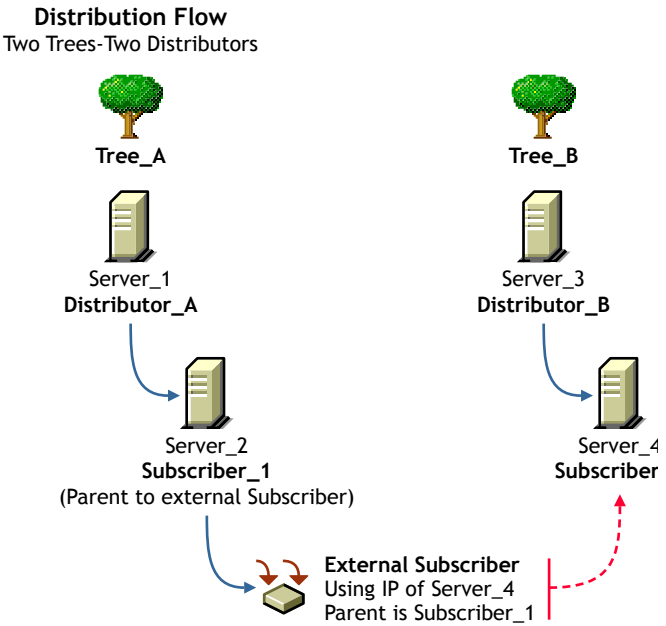
From an eDirectory perspective, the Distribution is sent from the Distributor object to Subscriber_1, which sends it to External Subscriber, which in turn sends it to Server_3. From a topology perspective, the Distribution file is sent from Server_1 to Server_2, which in turn sends it to Server_3.

External Subscribers, Multiple Distributors, and Multiple Trees

If you have two trees where Distributors have been installed, your network topology might be:



You could have the following TED configuration for Distributor_A's routing hierarchy:



In this example, each tree has a Distributor. Server_4 receives its configuration information from Distributor_B (Server_3) in its trusted tree. Therefore, a TEDNODES.PROPERTIES file is not needed for Server_4.

To send a Distribution from Distributor_A to Subscriber_2, you would create an External Subscriber object in Tree_A, select Subscriber_1 as the parent for the External Subscriber, and list Server_4's IP address in the External Subscriber object's properties.

From an eDirectory perspective, the Distribution is sent from Distributor_A to Subscriber_1, which sends it to External Subscriber. From a topology perspective, the Distribution file is sent from Server_1 to Server_2, which in turn sends it to Server_4.

Understanding External Subscribers

In summary, the External Subscriber object is useful for sending tree-to-tree Distributions when one of the following conditions exists:

- A target server has only the Subscriber software installed on it and no Subscriber object in any tree

- A target Subscriber server (object and software are installed) is in a different tree than the Distributor

The following are some points concerning External Subscribers:

- The main difference between the Subscriber and External Subscriber objects is that a Subscriber object is configurable and an External Subscriber object is not.
- An External Subscriber is typically used for getting Distributions from the Distributor in its own tree to a Subscriber in a different tree.
- An External Subscriber is typically a server running the Subscriber software, but does not have a Subscriber object in the same eDirectory tree as the Distributor from which it will receive a Distribution.
- An External Subscriber can be used for occasionally sending a Distribution to another tree, instead of duplicating the Distribution creation work in the other 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.
- A Subscriber receives its configuration information only from a Distributor in its trusted tree (the tree where the Subscriber object was created during installation). Therefore, if you have a Subscriber object in a different tree than the Distributor object, you can use an External Subscriber object in the Distributor's tree for sending the Distribution to that Subscriber. A TEDNODES.PROPERTIES file on the External Subscriber's target server would contain the configuration information for the Subscriber software.
- The Subscriber's information is stored in its eDirectory object, which is not accessible to TED agents in another tree. Therefore, the Subscriber object must have been configured by a Distributor in its own tree before it can receive a Distribution from a Distributor in another tree via the External Subscriber.
- A Distributor cannot send its Distributions to a server whose Subscriber object is in another tree. Therefore, in order for a server to receive Distributions from a Distributor, that server must be represented by either a Subscriber or External Subscriber object in the Distributor's tree.
- A parent Subscriber must be used for distributing tree-to-tree.
- An External Subscriber object must be associated with a parent Subscriber in its own tree, because a Subscriber in one tree that is associated with an External Subscriber object in another tree cannot receive Distributions directly from a Distributor in the External Subscriber's tree.

- 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.
- An External Subscriber must be subscribed to the Channel for the Distributions the Subscriber in the other tree needs.
- An External Subscriber is associated with a Subscriber using that Subscriber's IP address.
- External Subscriber objects cannot be used as parent Subscribers for Subscriber servers in another tree.
- If you install the Subscriber software without creating the Subscriber object (using the Installation Wizard), the TEDNODE.PROPERTIES file is automatically created and configured.
- A sample of the TEDNODE.PROPERTIES file is located on the *ZENworks for Servers Companion* CD under ZFS\TEDPOL\SFILES\TED. You can use this sample file to create a copy to place in the ZENWORKS\PDS\TED directory on the External Subscriber's server. For more information on configuring the TEDNODE.PROPERTIES file, see "Editing the TEDNODE.PROPERTIES File" on page 586.

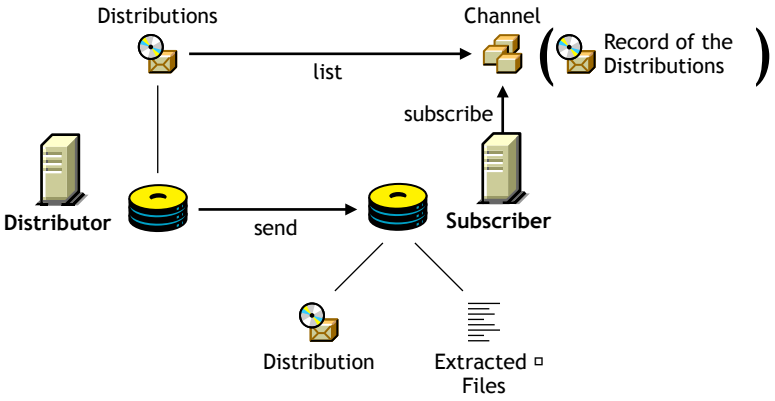
Understanding Distributor Routing Hierarchies

Review the following to understand how Distributions are sent:

- "Distribution Flow Details" on page 517
- "Distribution Routes" on page 517
- "Routing Hierarchy Guidelines" on page 521
- "Multiple Distributors and Parent Subscribers" on page 523
- "Understanding Distribution Routes" on page 525

Distribution Flow Details

The following illustrates the flow of TED Distributions:



Note the following from the illustration:

- A Distribution is stored on the Distributor server's hard drive
- The Channel lists a Distribution—it does not hold a copy of the Distribution—it only triggers sending the Distributions
- The Subscriber subscribes to a Channel to obtain the Distribution
- The Subscriber extracts the Distribution from its format and writes the files to the volume and directory specified when the Distribution was created

IMPORTANT: When there are multiple versions of a File or Desktop Application type of Distribution, the Subscriber maintains copies of the versions according to what you set in the Distribution object's properties. The default is 10.

Distribution Routes

TED provides a way to automate getting Distributions from the Distributors to your servers. The following sections provide information on sending Distributions:

- "Tiered Distribution Model" on page 518
- "Distributors Send Distributions Using Parent Subscribers" on page 518
- "Passing on Unsubscribed Distributions" on page 518
- "Routing Hierarchy" on page 519
- "Sharing the Distribution Load" on page 519

- "Distributors Can Share Distribution Routes" on page 520
- "Distributing Using the Hierarchy" on page 520
- "Orphaned Subscribers" on page 520
- "Rerouting a Distribution" on page 521

Tiered Distribution Model

The power of a tiered distribution model enables you to 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.

Distributors Send Distributions Using Parent Subscribers

A Distributor must actually send each of its Distributions because they reside on its own server.

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 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.

A detailed understanding of your network's topology is important for properly configuring distribution routes and selecting parent Subscribers. Therefore, refer to a diagram of your network that shows all WAN links.

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.

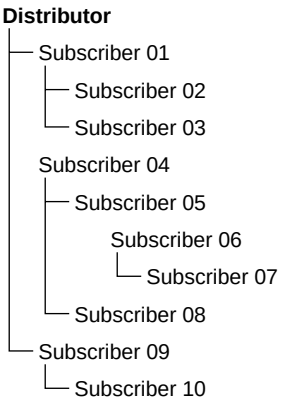
Routing Hierarchy

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.

Sharing the Distribution Load

In the illustration under "Routing Hierarchy" on page 519, 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).

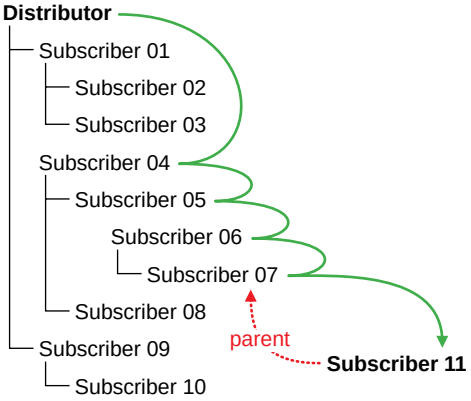
To further illustrate, parent Subscriber 04 would be servicing 104 Subscribers while only directly sending to two parent Subscribers (05 and 06) and its own 20 end-node Subscribers.

Distributors Can Share Distribution Routes

If you have multiple Distributors, they can share portions of each other's distribution routes, meaning Subscribers can be listed in the distribution routes 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.

Distributing Using the Hierarchy

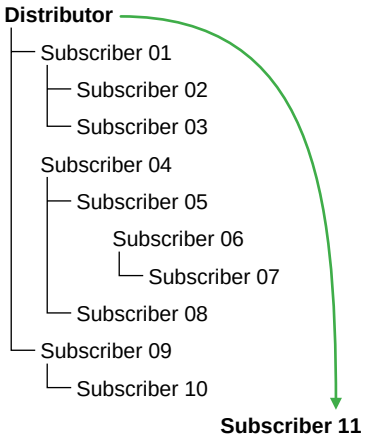
Assume that Subscriber 07 is a parent to Subscriber 11 (which is not in the routing hierarchy). The distribution route from the Distributor to Subscriber 11 would be the following:



The Distributor used four parent Subscribers (04, 05, 06, and 07) to send the Distribution to Subscriber 11.

Orphaned Subscribers

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 a Distribution

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.

Routing Hierarchy 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

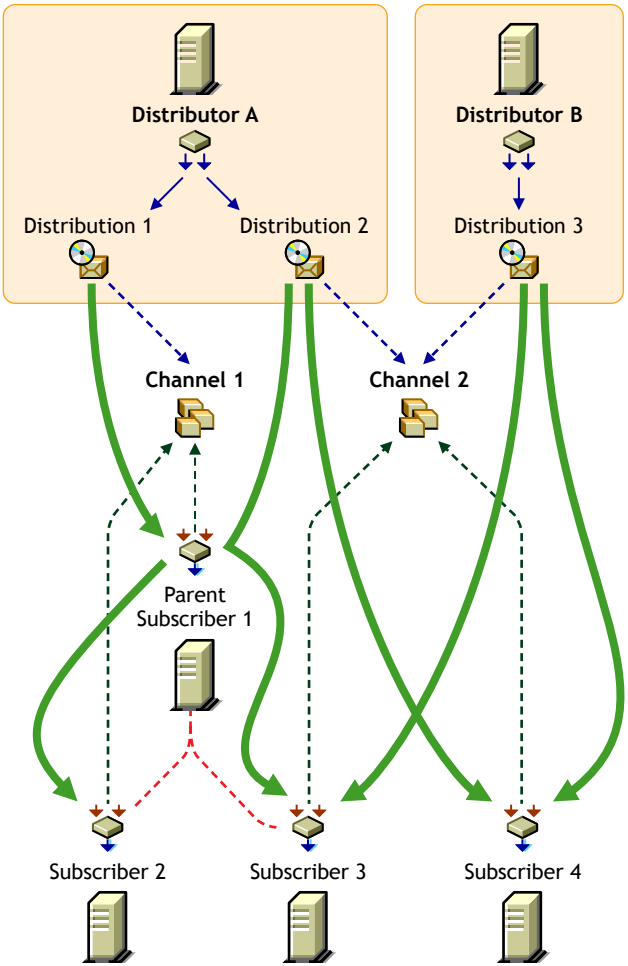
Note that parent Subscribers are not always required for a WAN link. For example, if you have only two Subscribers on a LAN connected by a fast WAN link, the traffic difference between sending the Distribution once versus twice could be negligible. However, for a slow WAN link this might not be the case.

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

Multiple Distributors and Parent Subscribers

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

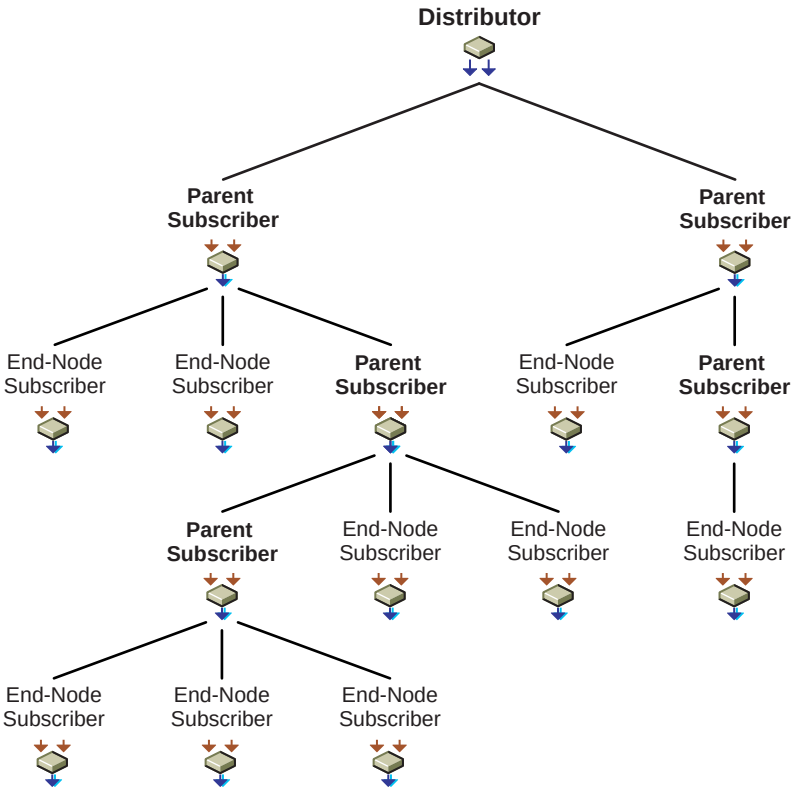
- Subscriber 1 is a parent to Subscribers 2 and 3
- Subscribers 3 and 4 are in Distributor B's hierarchy
- Subscriber 4 is not in Distributor A's hierarchy

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.

Understanding Distribution Routes

The following illustrates a Distribution hierarchy containing a Distributor, several parent Subscribers, and many end-node Subscribers:



Consider the following in designing your Distributor's routing hierarchy:

- **Distribution Route:** This 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 by being able to pass on Distributions.
- **End-Node Subscribers:** The only Subscribers you need to add to the left box are those you want to be used to pass on Distributions (see the Parent Subscriber nodes in the illustration). End-node Subscribers that will only receive Distributions 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 build the routing hierarchy to mimic your network topology.

- **Selecting Multiple Subscribers:** If you select multiple Subscribers, they will all be placed at an equal level under the Distributor or Subscriber you have selected in the left box.
- **Using Multiple Distributors:** You can also select multiple Distributors so that you can place the same Subscribers under each of them. This can be used to create the same Distribution route for each Distributor.
- **Reusing Subscribers:** Any Subscriber can be used more than once if you have multiple Distributors, although you should consider whether you might overload the Subscriber if it should service multiple Distributors.

TED Issues

- "Controlling I/O Rates and Concurrent Distributions" on page 526
- "Prioritizing Distributions" on page 527
- "Deleting a Distributor Object and How Its Distributions Are Affected" on page 528
- "Understanding Dependencies in TED" on page 530
- "System Resources and Server Behavior" on page 530
- "When a TED Process Fails" on page 532
- "Distribution Security" on page 533

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 527.

- **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.

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 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 (which Subscriber servers it used for sending its Distributions)
- The association with its Distributions (they become orphaned and unusable)

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 491.

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.

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 530
- "Unloading Parent Subscribers" on page 530

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.

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.

Distribution Security

Policy and Distribution Services provides several means for securing Distributions:

- “Certificates” on page 533
- “Encryption” on page 533
- “Inter-Server Communications” on page 533

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 695.

Encryption

Distributions can be encrypted for when you send them outside your secure network.

For more information, see “Distribution Security Using Encryption” on page 702.

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 707.

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 534
- "NetWare Subscriber Directories" on page 536
- "Windows NT Distributor Directories" on page 537
- "Windows NT Subscriber Directories" on page 537
- "UNIX Distributor Directories" on page 537
- "UNIX Subscriber Directories" on page 538

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_Distributio*n. 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.
digest_file	This file will only exist if the Distributor Agent creates 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. The syntax for creating the digest filename is: %AGENT%AgentDigest.TED For example: FTPAgentDigest.TED HTTPAgentDigest.TED FileAgentDigest.TED CPKAgentDigest.TED

NetWare Subscriber Directories

The following directories are used by NetWare Subscribers:

***volume:\installation_path*\ZENWORKS\PDS\TED**

Contains the TED software for the Subscriber and/or Distributor.

***volume:\installation_path*\ZENWORKS\PDS\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_Distributio*n. 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.

Filename	Description
digest_file	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\PDS\TED**

Contains the TED software for the Distributor.

***installation_path*\ZENWORKS\PDS\TED\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\PDS\TED\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.

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 type of 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 type of 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.

Configuring TED Objects

To set up TED, you must install TED objects on the servers across your tree. The installation program creates Distributor and Subscriber objects as part of the Policy and Distribution Services installation process. All other TED objects (Channels, Distributions, and External Subscribers) can be created in ConsoleOne.

After setting up the main TED objects (Distributors and Subscribers), you can then create Distributions, assign them to Channels, and associate Subscribers with specific Channels.

Do the following in order for each Distributor object:

- “Tiered Electronic Distribution Policy” on page 540
- “Configuring Distributors” on page 545
- “Configuring Subscribers” on page 549
- “Creating and Configuring External Subscribers” on page 553
- “Creating and Configuring Distributions” on page 554
- “Creating and Configuring Channels” on page 572

Tiered Electronic Distribution Policy

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 ZENworks Database 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 527.

   Maximum Concurrent Distributions: 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 533.

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: Check this option and enter the log file's filename using the following format:

installation_path\directory_path\filename.TXT

For example:

```
SYS:\ZENWORKS\PDS\LOGFILE\LOG.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.

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.
```

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 725
"Daily" on page 724
"Monthly" on page 725
"Yearly" on page 727
"Interval" on page 724
"Time" on page 726

For more information on the refresh schedule, see "TED Object Schedules" on page 714.

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 three minutes.

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 725
"Daily" on page 724
"Monthly" on page 725
"Yearly" on page 727
"Interval" on page 724
"Time" on page 726
"Run Immediately" on page 726

For more information on the extraction schedule, see "TED Object Schedules" on page 714.

8 Click OK to close the policy.

Configuring Distributors

You can edit your Distributor object's properties.

Distributor objects are automatically created when you the Distributor's software is installed to a server.

Not all properties associated with the Distributor object are required. Required properties are noted; all others are optional.

To edit the Distributor object's properties:

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 527.

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 533.

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 the log file's name and location. For example:

```
ZENWORKS\PDS\TED\DIST\DISTRIBUTOR.LOG
```

The default volume is SYS: on NetWare servers. We recommend that you do not use the SYS: volume because the log file can become quite large.

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 three minutes.

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 22, "Scheduling," on page 713.

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.

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 527.

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 533.

Parent Subscriber: Specifies a parent Subscriber from which all Distributions for this Subscriber will be received.

Because the routing information in a Distributor object's properties only accounts for the parent Subscribers, this field is where you can connect end-node Subscribers into a Distributor's distribution route. End-node Subscribers are those that will not be used to pass Distributions to other Subscribers.

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 the log file's name and location. For example:

```
ZENWORKS\PDS\TED\SUB\SUBSCRIBER.LOG
```

The default volume is SYS: on NetWare servers. We recommend that you do not use the SYS: volume.

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 22, "Scheduling," on page 713.

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 more information on variables, see "Using Variables to Control File Extraction" on page 738.

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.

Creating and Configuring External Subscribers

You can create External Subscriber objects for sending Distributions to Subscribers residing on other trees.

The following sections provide steps to create and configure an External Subscriber:

- "Creating an External Subscriber Object" on page 553
- "Configuring the External Subscriber Object" on page 553

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.

Creating and Configuring Distributions

The following sections provide information on creating and configuring Distributions:

- "Understanding Distributions" on page 555
- "Determining the Distributions" on page 557
- "Creating and Distributing New Versions of Existing Distributions" on page 558
- "Creating a Distribution" on page 559

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 555
- "I/O Rate (Bytes per Second)" on page 556
- "Updating the Distributor's eDirectory Information" on page 556
- "Checking the Distribution Package Changes" on page 556
- "Determining the Sizes and Frequencies for Distribution Packages" on page 556
- "File Type of Distribution" on page 557

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.

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 490).

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 Sizes and Frequencies for Distribution Packages

The size and frequency of a distribution 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 type of 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.

File Type of Distribution

The File type of Distribution sends only the differences between the previous version of the Distribution and the current version on the Distributor or parent Subscriber. There is no option to send the entire 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.

The unique feature of the File type of Distribution is that it calculates the differences at the time the Distribution is built and only builds a delta of it.

The File type of Distribution is a good means of distributing large Distributions that change often (so updates are distributed frequently). It is also effective when changes are substantial because it can build much smaller deltas.

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 AUTOEXEC.NCF 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
```

If you later decide not to create a Distribution, you can delete the related Channel object.

Creating and Distributing New Versions of Existing Distributions

The following sections provide information on build schedules and maximum revisions:

- "Build Schedule" on page 558
- "Maximum Revisions" on page 558

Build Schedule

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 type of 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.

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 type of 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 type of 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.

Creating a Distribution

There are several TED Distribution types. Each type has unique features that tailor it for specific needs.

The following sections explain each Distribution type and provide instructions on creating a Distribution:

- "File" on page 559
- "HTTP, FTP, and RPM" on page 560
- "Software Package" on page 560
- "Policy Package" on page 560
- "Desktop Application" on page 560
- "Creating the Distribution" on page 560

File

The Distributor will build the entire Distribution (baseline) for the first Distribution. For all subsequent versions it will create a delta file. The delta contains only the changes between the current version and the previous version. The delta file is what is distributed to Subscribers—not the entire Distribution.

When the maximum number of revisions has been reached, the Distributor will delete all previous versions and build a new complete Distribution (baseline). The File type of Distribution is a useful means of distributing large Distributions that change often and have changes that can be substantial.

HTTP, FTP, and RPM

The Distributor will build the entire Distribution each time the Build schedule starts. It will send each new version of the Distribution to the appropriate Subscribers.

Software Package

Software Package is the most robust Distribution type. 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. For more information, see Chapter 19, "Server Software Packages," on page 641.

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, policies are now distributed for enforcement on the receiving Subscriber servers.

Desktop Application

This Distribution type is used in conjunction with ZfD to distribute desktop applications. It automatically distributes a copy of the original Desktop Application object along with its desktop application files from one location in the eDirectory tree to another location, without manually copying files down to servers in the physical area of the eDirectory tree. It performs all of the appropriate hookups to the Desktop Application object to render it fully functional.

Creating the Distribution

To create a Distribution object:

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. Click either Strong or Weak encryption. Encryption is not available for Linux or Solaris servers.

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 type of 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 more information on prioritizing Distributions, see "Prioritizing Distributions" on page 527.

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	The target file system's location for where you want the Distribution to be extracted. %DEST VOLUME% is the default. You can use any type of variable, or the actual location names. For example: **NetWare:** SYS:\FILES DATA:\FILES **Windows:** C:\FILES \\MyServer\Files *(shared folder)* **Linux or Solaris:** /usr/files Do not use a UNC path or all Distributions will be sent to that one location.
Add Directory	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. 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.

Button	Explanation
Add Files	Browse for directories or files on the Distributor's file system that you want copied to the target Subscriber's file system. 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. 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.
Delete	Only deletes whatever you have selected from the tree structure in the Files To Be Distributed box. ◆ **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	This causes the directories on the target server to be synchronized with the directories contained in the Distribution. **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. 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. 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.
Verification Distributions	Each time a Distribution changes, such as files are modified or added, a new version is built and subsequently sent to the Subscribers. However, Subscribes 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. 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.
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.

Field	Explanation
Kill Conn on 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. Also, server and NLM™ connections cannot be killed.
Maintain Trustees	Maintains each file's trustee attributes for the target file system as they are on the source file system. This is additive, meaning it will not remove trustees on the target file system.
Error Handling	You have two options: ◆ **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.

Button	Explanation
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: ◆ **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.
Delete	Deletes whatever you have selected: ◆ **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.

Button	Explanation
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, see "Converting Older Server Software Packages to ZfS 3" on page 679.

  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 more information on creating specific policies, see Chapter 18, "Server Policies," on page 591.

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.

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 572
- "Creating the Channels" on page 573
- "Configuring the Channels" on page 573

Determining the Channel Names

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

Creating the Channels

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 "Creating a Distribution" on page 559.
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 22, "Scheduling," on page 713.

Creating Security Certificates

There are two features of TED that deal with security:

- **Certificates:** Certificates (required) 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 a certificate in its security directory from that Distributor.
- **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.

Important points about certificates:

- Security: Certificate key pairs are created by the Distributor.
- The public key is written to the Distributor server's file system, which self-assigns a certificate and stores it in eDirectory.
- A certificate can be passed from a Distributor to a Subscriber under two circumstances:
  - A Subscriber subscribes to a Channel
  - A Distribution is listed in a Channel

  In both cases, a message will be displayed asking whether to resolve certificates.
- Subscribers do not need to be running to have certificates copied to their servers.
- 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 form the Subscriber's security subdirectory. Then the Distributor must send the new certificates to those Subscribers.

Important points about digests:

- The Digest option is for all Distribution types. The Digest check box is displayed on the General tab of the Distribution object.

- A digest is created at the time a Distribution is built. It is used by a Subscriber to determine whether a Distribution has been tampered with after it left the Distributor.
- 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.

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.
   - 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.

Scheduling and Sending Distributions

When a Channel schedule starts, the Distributors that own Distributions associated with the Channel will send their Distributions to the Subscribers associated with that Channel.

The following sections provide information on scheduling and sending Distributions:

- "Setting Up Distribution Schedules" on page 576
- "Associating Subscribers with Channels" on page 579
- "Preparation Checklist for Sending Distributions" on page 579

Setting Up Distribution 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.

The following sections provide information on setting up your TED object schedules:

- "Understanding Scheduling in Policy and Distribution Services" on page 576
- "Scheduling the Distributor" on page 577
- "Scheduling Channels" on page 577
- "Scheduling Subscribers" on page 578
- "Scheduling the Distribution" on page 578

Understanding Scheduling in Policy and Distribution Services

Schedules can be defined with user-defined windows of opportunity during the day for the schedule's action to begin.

The Action schedule triggers when a process is to occur, such as updating a Distribution version. Action schedules have an option for randomly dispatching that 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.

Scheduling does not affect the total resources used by a Distribution, just when the resources will be used.

Distributions are anticipated to occur during off-peak hours. For some networks, it is possible that this 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.

For more information on scheduling, see Chapter 22, "Scheduling," on page 713.

Scheduling the Distributor

A Distributor's schedule determines when the Distributor will re-read eDirectory for updates.

To schedule the 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 every 5 minutes) > click OK.

Scheduling Channels

A Channel's schedule determines when the Distributor starts to send its Distributions to the Subscribers associated with the Channel.

A Distributor distributing to the first tier (Subscribers getting Distributions directly from the Distributor) will stop sending the Distribution when the Channel schedule ends. It will start again the next time the Channel schedule starts. The Distributor will pick up where it left off.

Second-tier receivers and beyond will 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 Channel schedule ends. Their subordinate Subscribers will also ignore the Channel schedule.

For information on how time zones affect a Channel's schedule, see "TED Object Scheduling Issues" on page 720.

To schedule the 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 (such as every 5 minutes) > click OK.

Scheduling Subscribers

The Subscriber's schedule determines when the Subscriber will extract Distributions.

To schedule the 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 > Run Immediately > OK.

   This schedule type will cause the Subscriber to extract the Distribution as soon as it is received.
4. Repeat these steps for each Subscriber.

Scheduling the Distribution

The Distribution's schedule determines when a Distributor will be requested to build the individual pieces that comprise the Distribution.

To schedule the Distribution:

1. In ConsoleOne, right-click a Subscriber object > click Properties.
2. Click the Channels tab > Add.
3. 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.
4. Click Schedule > click the arrow for the drop-down list > select a schedule (such as Run Immediately).

   This will cause the Distributor to build the distribution according to the schedule as soon as it re-reads the Distribution information from eDirectory.
5. Click OK > Yes (to resolve certificates).

   This will copy the security certificates from the Distributor to the Subscribers that are subscribed to the Channel. For more information on certificates, see "Creating Security Certificates" on page 574.

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 22, "Scheduling," on page 713.
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 more information on variables, see "Using Variables to Control File Extraction" on page 738.

Preparation Checklist for Sending Distributions

The steps in this section will help you to understand the process of creating and sending a Distribution. Up to this point you should have:

- Configured the necessary policies
- Configured and scheduled the Distributors
- Configured and scheduled the Subscribers
- Created, configured, and scheduled the Distributions
- Created, configured, and scheduled the Channels

If you have not completed all of the above items, return to the appropriate instructions and perform the necessary tasks.

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 it already knows about.

For example, you create a new Distribution. The Build schedule will not make the Distributor aware of the new Distribution. You must manually refresh the Distributor.

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 490.

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.

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.

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.

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 more 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 553.

3. In TREE1, create the Channel for the Distribution.

   For steps in creating Channels, see "Creating and Configuring Channels" on page 572.

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 579.

5. In TREE1, create the Distribution.

   For steps in creating Distributions, see "Creating and Configuring Distributions" on page 554.

6. Associate this Distribution with the Channel you created in Step 3.

7. Verify that the External Subscriber server in TREE2 received the Distribution.

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

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 547
- **Subscriber object:** Step 3 on page 551

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 545
- **Subscriber object:** Step 2 on page 549

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 547
- **Subscriber object:** Step 3 on page 551

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.

Manually Importing/Exporting Distributions

You can manually import or export a Distribution from and to a media source, such as a floppy disk or ZIP drive. Exported Distributions use the .TED extension.

The following sections provide information on exporting and importing Distributions:

- "Exporting a Distribution" on page 585
- "Importing a Distribution" on page 585

Exporting a Distribution

To export a Distribution:

1. In ConsoleOne, click Tools > TED Manual Distribution to start the Manual Distribution Wizard.
2. Click the Export radio button > click Next.
3. Select a Channel > select a Distribution from that Channel > click Next.
4. Enter a path and filename for the Distribution > click Next.

   The filename should have .TED as its extension.
5. For Linux and Solaris servers, map a drive to the path you specified in Step 4.
6. If you are satisfied with the summary, click Finish.

   The Distribution file is saved to the media source you specified.

   For Linux or Solaris servers, you will be asked to browse for the working directory, which is on the drive you mapped in Step 5.

Importing a Distribution

To export a Distribution:

1. In ConsoleOne, click Tools > TED Manual Distribution to start the Manual Distribution Wizard.
2. Click the Import radio button > click Next.
3. Enter the path and filename to the Distribution file > click Next.

   The filename should have .TED as its extension.

4 Select parent Subscribers in the top box and/or individual Subscribers in the bottom box > click Next.

If you select a parent Subscriber in the routing hierarchy, all the Subscribers below it in the hierarchy will receive the Distribution, if they are already subscribed to the Channel.

The Subscribers displayed in the bottom box are only those who are currently subscribed to this Channel. The heading displays the Channel that is associated with the Distribution being imported (which information is contained in the .TED file).

External Subscribers will not be listed in the bottom box because they cannot receive manual Distributions.

5 For Linux and Solaris servers, map a drive to the working directory you specified in Step 3.

6 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 Subscribers.

For Linux or Solaris servers, you will be asked to browse for the working directory, which is on the drive you mapped in Step 5. If you are not properly authenticated to a Windows server, you can receive this same request to browse for the working directory.

Editing the TEDNODE.PROPERTIES File

If an External Subscriber server is not set up as a Subscriber object in any eDirectory tree, you can place a TEDNODE.PROPERTIES file on that server so that it can perform Subscriber functions. The TEDNODE.PROPERTIES file must be placed in the ZENWORKS\PDS\TED directory on the server.

A sample of the TEDNODE.PROPERTIES file is located at the root of the *ZENworks for Servers Companion* CD.

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 file name
workorder.timeout = 0	Number of seconds to wait for reply from the Distributor before dropping connection; 0 = wait forever
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
Interval schedule and associated keys (with default values specified) #schedule.type=interval #schedule.repeat.hours=0 #schedule.repeat.minutes=0

Line Content
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

18 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 591
- "Creating a Policy Package" on page 606
- "Configuring Server Policies" on page 608
- "Enabling Policies" on page 633
- "Distributing Policies" on page 633
- "Associating Policies" on page 634
- "Scheduling Policies" on page 635
- "Viewing Effective Policies" on page 636
- "Changing Policy Enforcement" on page 637

Understanding Server Policies

In ZfS 3, 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, all policies were enforced through container and object associations.

Review the following sections to understand polices in ZfS 3:

- "Configuration and Behavioral Management through Server Policies" on page 592
- "Server Policies and Packages" on page 593

- "Plural and Cumulative Policies" on page 593
- "Server Policies Architecture" on page 594
- "Configuration and Behavioral Policies" on page 594
- "Enforcing Policies" on page 598
- "Server Policy Descriptions" on page 600

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 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 593
- "Cumulative Policies" on page 594

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 595
- "Policy/Package Agent" on page 598

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 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 servers)

The Server Package provides backwards compatibility that allows you to run ZfS 3 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 596
- "Server Package Object" on page 596
- "Service Location Package Object" on page 596
- "Distributed Server Package" on page 597

- "ZENworks Database Object" on page 597

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) 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 Server Inventory:

Copy Files (ZfS 3 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 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 Distributors and Subscribers.

Distributed Server Package

The Distributed Server Package object is an eDirectory object that manages the following policy objects (ZfS 3 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 type of 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 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 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 599
- "Distributing Policies" on page 599
- "Associating Policies" on page 600

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 600
- "Service Location Package" on page 601
- "Server Package" on page 603
- "Distributed Server Package" on page 604

Container Package

This policy description only applies to ZfS 3. 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.

Policy Name	Policy Type Keys	Policy Function
Search	Behavioral	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. This Search policy can only be administered in ConsoleOne®. A Search policy created in NetWare Administrator for ZENworks will not be recognized in ZfS.

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.

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	Sets SNMP trap targets for associated eDirectory objects. In ZfS 3, 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. IPX™ addresses are not supported for SNMP trap targets. Only IP addresses and DNS names can be used.

Policy Name	Policy Type Keys	Policy Function
Tiered Electronic Distribution	Configuration	Sets defaults for the Distributor and Subscriber objects, including: 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 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.
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 744.
ZENworks for Servers License	Configuration	ZfS 2 only. Identifies the NLS object, otherwise ZfS 2 Policy and Distribution Services will not work.

Server Package

The Server Package exists in ZfS 3 for backwards compatibility with ZfS 2, such as when upgrading incrementally. This package also exists to provide polices that must be associated, such as for ZfS 3 Server Inventory or ZENworks for Desktops (ZfD) 3.*x*. 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 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 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 servers can use this policy.
- The SNMP Trap Target Refresh policy will not display if only ZfS 3 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 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 Server Package, you must have done the following during upgrading:

1. Updated the ConsoleOne version that ZfS 2 is using by installing version 1.3.3 over it from the ZfS 3 *Companion* CD.
2. Installed the ZfS 3 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 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 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.

Although the ZENworks Database policy did not exist in ZfS 2, it will be displayed in this package. Only the ZfS 3 Server Inventory component uses the ZENworks Database policy. For more information, see “Configuring the Database Location Policy” on page 858.

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).

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 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.
Server Scripts	Plural Cumulative Configuration	For automating script usage on your servers.

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 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 744.

Policy Name	Policy Type Keys	Policy Function
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 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 607
- “Creating a Policy Package Object” on page 607

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 608
- "Configuring the Container Package Policy" on page 608
- "Configuring Server Package Policies" on page 611
- "Configuring Service Location Package Policies" on page 611

For information on scheduling server policies, see "Scheduling Policies" on page 635.

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 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 *ZENworks for Servers* product CD at: \ZfS\TEDPOL\SFILES\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
[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 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 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 Policy and Distribution Services servers. For more information, see "Configuring Distributed Server Package Policies" on page 617.

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).

For ZfS 3, only Server Inventory uses the ZENworks Database policy contained in the Server Package. For more information, see "Configuring the Database Location Policy" on page 858.

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).

To configure Service Location Package policies, review the following sections:

- "SMTP Host" on page 611
- "SNMP Trap Targets" on page 612
- "Tiered Electronic Distribution" on page 613
- "ZENworks Database" on page 616

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, 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 724
"Event" on page 724
"Interval" on page 724
"Never" on page 725
"Package Schedule" on page 725
"Relative" on page 726
"Run Immediately" on page 726
"Time" on page 726
"Weekly" on page 727
"Monthly" on page 725
"Yearly" on page 727

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

The Tiered Electronic Distribution policy sets defaults for the Distributor and Subscriber objects.

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:

I/O Rate: Sets the default I/O rate to minimize network traffic for TED objects.

Output Rates Based Upon Distribution's Priority: The three priority options (High, Medium, or Low) determine when the Distribution will be sent. For more information, see "Prioritizing Distributions" on page 527.

Maximum Concurrent Distributions: Specifies a default maximum number of Distribution threads that can be running concurrently. The default value is 10.

Connection Time-out: Specifies a default allotment of time before the Distributor or Subscriber times out when connecting to another node. The default value is 1 minute.

This setting can be increased or decreased to allow for messages to pass back and forth between agents during the Distribution process. If one node is expecting to receive a message from another there needs be a reasonable time to wait before giving up and assuming that the sender is no longer coherent.

Working Directory: Enter a default TED directory to store archives, persistent status, and temporary files on a server. The directory needs to be located on a volume with enough free space to handle processing of Distributions.

Parent Subscriber: If Subscribers are generally not to receive Distributions directly from a Distributor, you can browse for a default parent Subscriber that will pass on the Distributions indirectly from the Distributor.

Disk Space Desired To Be Left Free: Use this value to ensure there will be enough free disk space for receiving Distributions.

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: Check this option and enter the log file's filename. For example, SYS:\ZENWORKS\PDS\TED\LOGFILES\TEDLOG.LOG. Do not include the machine's name. This default log file would be used by both Distributors and Subscribers.

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 the following options:

CN	Generational Qualifier
Description	Given Name
EMail Address	Initials
Full Name	Internet EMail Address

5 To assign default variables (for Subscribers only), click the Variables tab > click Add > fill in the fields:

Variable: The variable name can be derived from predefined and user-defined variables.

Value: 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 the variable.

IMPORTANT: Variables set on a Subscriber object will override the values you set in this policy.

For more information on creating variables for Subscribers, see "Creating a Variable" on page 735.

6 To assign a default refresh schedule for all Distributors for when they should re-read eDirectory, click the Schedule tab > click Distributor Refresh Schedule > select a schedule in the Schedule Type field > configure the schedule:

"Never" on page 725
"Daily" on page 724
"Monthly" on page 725
"Yearly" on page 727
"Interval" on page 724
"Time" on page 726

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 three minutes.

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 725
"Daily" on page 724
"Monthly" on page 725
"Yearly" on page 727
"Interval" on page 724
"Time" on page 726
"Run Immediately" on page 726

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 618
- "NetWare SET Parameters" on page 619
- "Scheduled Down" on page 621
- "Scheduled Load/Unload" on page 622
- "Server Down Process" on page 623
- "Server Scripts" on page 625

- “SMTP Host” on page 626
- “SNMP Community Strings” on page 626
- “SNMP Trap Targets” on page 627
- “Text File Changes” on page 629
- “ZENworks Database” on page 630
- “ZENworks for Servers” on page 631

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 635).

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 635).

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 635).
8. Click OK to close the policy.

   If you click Cancel, neither schedule for your new 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 1097
   "Load Java Class" on page 1098
   "Unload Process" on page 1098
   "Start Service" on page 1098
   "Stop Service" on page 1099

   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 635).
11. Click OK to close the policy.

    If you click Cancel, your new 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 608.

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 635).

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 635).
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 608.

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 or 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.

HINT: 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 635).

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 635).
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 608.

4c Select the procedure to follow when recording information to a log file.

Logging Procedure	Description
Log File	Check to enable and enter the log file's filename. Include its full path. If you don't give a filename for the log file, Policy and Distribution Services uses ZFSLOG.TXT as the default and places it under the ZENWORKS\PDS\SMANAGER directory. Until Policy and Distribution Services has loaded and read eDirectory, it will temporarily use ZFSINIT.TXT for logging.

Logging Procedure	Description
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.
◆ 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 732).

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.

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 633.

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 the Policy Package type of Distribution.
2. Configure the policies in the policy package.
3. Select a Channel for the Policy Package Distribution.
4. Subscriber 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 polices 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 611).

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 634
- "Associating the Distributor Object to a Policy Package" on page 634

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 635
- “Editing the Default Schedule” on page 635

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 724
   “Event” on page 724
   “Interval” on page 724
   “Never” on page 725
   “Package Schedule” on page 725
   “Relative” on page 726
   “Run Immediately” on page 726
   “Time” on page 726
   “Weekly” on page 727
   “Monthly” on page 725
   “Yearly” on page 727

   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 724
"Event" on page 724
"Interval" on page 724
"Package Schedule" on page 725
"Relative" on page 726
"Run Immediately" on page 726
"Time" on page 726
"Weekly" on page 727
"Monthly" on page 725
"Yearly" on page 727

Viewing Effective Policies

The procedures for viewing which policies are in effect are different for ZfS 2 and ZfS 3:

- "Viewing Effective Policies for ZfS 3 Servers" on page 636
- "Viewing Effective Policies for ZfS 2 Servers" on page 636

Viewing Effective Policies for ZfS 3 Servers

To view which ZfS 3 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 637
- “Stopping a Specific Policy From Being Enforced” on page 637
- “Removing Policy Enforcement for a Specific Subscriber” on page 638
- “Stopping Enforcement of a Policy Package Type of Distribution” on page 639

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 Type of Distribution

If you need to stop enforcement of a Policy/Package type of 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 type of 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.

19 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 641
- "Understanding Server Software Packages" on page 642
- "Planning Server Software Packages" on page 649
- "Setting Up Server Software Packages" on page 658
- "Converting Older Server Software Packages to ZfS 3" on page 679

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 19, “Server Software Packages,” on page 641.

For information on converting older .SPK and .CPK files to ZfS 3, see “Converting Older Server Software Packages to ZfS 3” on page 679.

Understanding Server Software Packages

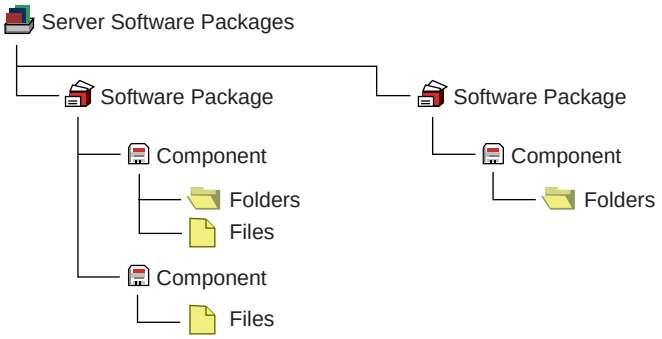
Policy and Distribution Services provides the means to automate and standardize the distribution and installation of server files and applications. This includes your ability to standardize NLM™ versions, configuration files, databases, and more. Review the following sections:

- “Understanding Server Software Packages and Components” on page 642
- “Understanding Software Package and Component Configurations” on page 643
- “Determining the Installation Order of Software Packages” on page 644
- “Compiling Software Packages” on page 645
- “Accessing Software Packages” on page 646
- “Distributing Software Packages” on page 647
- “Rolling Back Software Package Installations” on page 648

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 an .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

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 contained in one Distribution that start their installations in a certain order might not all finish in that same order

Therefore, if you have software packages that must be installed in a particular order, do the following:

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.

Compiling Software Packages

Once 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 E, "Using Server Software Packages to Delete Directories on Servers," on page 1113).

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 648.

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 646
- "Running ConsoleOne from a Server" on page 646

For more information on managing software packages from multiple workstations, see "What Are My Software Package Management Options?" on page 650.

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.

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 648, 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 481.

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 later 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 649
- "What Are the Software Package Components?" on page 649
- "What Are the Minimum Requirements?" on page 650
- "What Are My Software Package Management Options?" on page 650

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 ZfD to workstations. For more information on configuring a Desktop Application Distribution, see the Novell Documentation Web site (http://www.novell.com/documentation/lg/zdfs/index.html), and under Administration, click Application Management.

If you have ZENworks for Desktops (ZfD) installed, you can also distribute desktop applications using TED, instead of including them in software packages. For more information, see Chapter 20, "Desktop Application Distribution," on page 683.

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 663 and "Creating the Software Package Components" on page 665 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 665 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 664 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 651
- "Understanding Your Software Package Management Options" on page 652
- "Storing and Managing .SPK Files Using One Workstation" on page 652
- "Storing .SPK Files on a Network Server and Managing Them from Multiple Workstations" on page 652
- "Example in Using a Master SNAPINPREFS.SER File" on page 656

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 653
- "Managing Software Packages from Multiple Workstations" on page 654

- “General Rules for Managing Software Packages from Multiple Workstations” on page 655
- “The Best Scenario for Using Multiple Workstations to Manage Software Packages” on page 656

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 656.

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 658).
4. Create a batch file for starting and stopping ConsoleOne on a workstation (see "Creating the ConsoleOne Batch File" on page 660). 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 663).

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 656.

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 658
2. “Creating a Server Software Package” on page 663
3. “Configuring the Server Software Package” on page 664
4. “Creating the Software Package Components” on page 665
5. “Configuring the Software Package Components” on page 665
6. “Compiling a Software Package” on page 678
7. “Distributing the Software Package” on page 679

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 SNAPINSPREFS.SER File” on page 658
2. “Creating the ConsoleOne Batch File” on page 660
3. “Using the ConsoleOne Batch File” on page 663

Setting Up the Master SNAPINSPREFS.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 more information on creating software packages, see "Setting Up Server Software Packages" on page 658.

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 660.

Creating the ConsoleOne Batch File

Review the following sections to create and use the batch file:

- "Sample Batch File" on page 660
- "What the Batch File Does" on page 661
- "Creating Your Batch File" on page 661
- "Optional Modifications to the Batch File" on page 661

Sample Batch File

```
@echo off
REM map a network drive
net use m: \\prv-ale.provo.novell.com\vol1

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 658).
- 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 659.
- 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,

  `C1SSP 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 SNAPINPREFS.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 (.CPKs) can be in the hundreds of megabytes. For more information on storing .SPK and .CPK files, see "What Are My Software Package Management Options?" on page 650.

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 more information on rolling back software package installations, see "Rolling Back Software Package Installations" on page 648.

4 Click the Requirements tab.

5 Click Add > select a requirement:

"Operating System" on page 1102
"Memory (RAM)" on page 1103
"Disk Space" on page 1103
"SET Commands" on page 1104
"Registry" on page 1104
"File" on page 1105
"PRODUCTS.DAT" on page 1105

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 more information on variables, see "Using Variables to Control File Extraction" on page 738.

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 666
"Pre-Installation Load/Unload Order" on page 667
"Pre-Installation Scripts" on page 667
"Copy File" on page 669
"Text File Changes" on page 673
"SET Commands" on page 675
"Registry Settings" on page 675
"PRODUCTS.DAT" on page 676
"Post-Installation Unload/Load Order" on page 677
"Post-Installation Scripts" on page 678

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 678 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 1102
"Memory (RAM)" on page 1103
"Disk Space" on page 1103
"SET Commands" on page 1104
"Registry" on page 1104
"File" on page 1105
"PRODUCTS.DAT" on page 1105

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 1097
   "Load Java Class" on page 1098
   "Unload Process" on page 1098
   "Start Service" on page 1098
   "Stop Service" on page 1099

   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.

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.

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.

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 4.

If you selected a directory on the Open dialog box, it will be displayed unexpanded. 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:

1. While displaying the properties of the software package component, click the Registry Settings tab > click HKEY_LOCAL_MACHINE.
2. Click Add.

3 Select from the following:

"Key" on page 1108
"Binary" on page 1108
"Expand String" on page 1109
"(Default)" on page 1109
"DWord" on page 1110
"Multi-Value String" on page 1110
"String" on page 1110

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 1097
"Load Java Class" on page 1098
"Unload Process" on page 1098
"Start Service" on page 1098
"Stop Service" on page 1099

Click 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 (.SPKs) 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 (.CPKs) 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 679 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 A, "Server Console Commands," on page 1089 for instructions on using the command)
- Use TED (see Chapter 17, "Tiered Electronic Distribution," on page 495 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

ZfS provides a wizard for converting older Server Software Packages to ZfS 3. The conversion works for both ZfS 1.0 and ZfS 2 software packages.

This wizard does not ship on the *ZENworks for Servers* product CD. To obtain the wizard, search the Knowledgebase at Novell Technical Services (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:

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 files, a message will be displayed indicating the files are already version 3.

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 servers.

20 Desktop Application Distribution

Novell® ZENworks® for Servers (ZfS) Policy and Distribution Services provides integration with ZENworks for Desktops (ZfD) Novell Application Management.

ZfD can use Tiered Electronic Distribution (TED) to distribute application objects to other locations in the same tree or other trees. The distribution includes copying the original files associated with the applications to the appropriate server locations, where they can be used to service user groups and workstation groups associated with the distributed application objects.

To distribute ZfD applications, you use the Desktop Application Distribution Wizard to configure your Distribution. This includes:

- Determining the destination's tree structure
- 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
- Selecting your applications
- Determining your file copying paths

Once you have created the Desktop Application Distribution and it has been built on the Distributor, sent through a Channel, and extracted on the Subscriber, objects in your user groups, workstation groups, and containers at the destination location can use the distributed applications through ZfD.

User and workstation objects are not replicated with the user and workstation groups or containers. If the user group, workstation group, or container to be replicated exists in the target location, the wizard will not need to create them, and 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.

If the target destination does not have the groups or containers to be replicated, the Distribution extraction will create them. In that case, you will need to populate them with the users and workstations who need the distributed applications.

For information on interoperability issues between ZfS 3 and ZfD, see Interoperability Between ZENworks for Servers and ZENworks for Desktops in the *Installation* guide.

The following sections provide information on setting up and using the integration between ZfS and ZfD:

- "Requirements" on page 684
- "Creating a Desktop Application Distribution" on page 685
- "Rebuilding Desktop Application Distributions" on page 690
- "Linking Site Lists" on page 691

Requirements

The following requirements must be met before creating and distributing ZfD Desktop Application Distributions using TED. For previous ZfS 2 users, note that some of these requirements are different in ZfS 3.

- To use Novell Application Management with ZfS 3, ZfD 3.*x* or ZfD 4 is required.
- ZfD and ZfS must both be installed to the same tree, including their respective schema extensions.
- The ZfD application objects to be distributed must have been created in Novell eDirectory™ with ZfD and must be functional before creating the Desktop Application Distribution in ZfS 3.
- ZfS 3 does not use a power user to gain eDirectory access.
- The source path must be a valid UNC path that points to application files that must be located on the Distributor server's file system. If the Source Path field contains a local drive mapping, no application files will be gathered or distributed.
- ZfS 3 does not use a Site Distribution object.
- The Subscriber object must have the Working Context attribute defined. This is the eDirectory context where the Subscriber will create the objects related to the Desktop Application Distributions it receives.

- eDirectory must be installed on any Windows* Subscriber server where the Distribution will be sent and extracted.
- 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.

Creating a Desktop Application Distribution

To create a Distribution using the Desktop Application type:

1. In ConsoleOne®, right-click the container where you want the Distribution object located > click New > click Object > select TED Distribution > click OK.
2. 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.
3. 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.
4. 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 a checksum for the Subscriber to compare.

   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. Select between Strong and Weak encryption. Encryption is not available for Linux* or Solaris* servers.

   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 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.

Distributor: Displays the DN of the Distributor object that will build and send this Distribution. You selected the Distributor when you created the 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.

You can select whether to have platform restrictions for the Distribution.

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 where their 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 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 > in the Select Type drop-down box, select Desktop Application > click Setup.

The Desktop Application Distribution Wizard is started.

6a Click Next after reading the Introduction information.

6b 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 ZfD application objects. If you will be selecting chained applications, you must check this option.

Maintain Associations: Maintains the associations established in the source tree between the distributed applications and the trusted user/workstation groups and containers. This is done by replicating 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 replicated.

Source Root Container: Select a container to be used as the root container for the ZfD application objects to be distributed. You should only select 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, and the functionality of fault tolerance 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 has gone down. Fault Tolerance does not include load balancing. Select None to manually configure each application object for load balancing or fault tolerance.

6c Click Add to browse for and select ZfD 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.

6d Enter the destination volume or shared folder.

The application files distributed are those that are associated with the 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 or shared folder becomes the root location for placing subordinate directories where the application files will be copied.

6e To use only an application's default path, click Default Application Directory Path, which will be placed beginning with the root location you specified in Step 6d.

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 6d).
- 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.

6f Click Next to continue.

The Summary page is displayed.

6g To make changes, click Back.

6h When you have finished configuring the Distribution object, click Finish.

You can edit the Distribution at any time on the Type tab of the Distribution object by clicking Modify.

7 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 will be sent to all Subscribers that are subscribed to the selected Channel.

8 Click the Schedule tab > select a Build schedule:

"Never" on page 725
"Daily" on page 724
"Monthly" on page 725
"Yearly" on page 727
"Interval" on page 724
"Time" on page 726
"Run Immediately" on page 726

9 Click Apply to create the Distribution.

You will be prompted to copy additional security certificates.

10 Click Yes to resolve the certificates.

This will copy the security certificates from the Distributor to Subscriber that is subscribed to the Channel.

11 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 will retrieve all of the information about the new Desktop Application Distribution, such as Distribution details, the Build schedule, and so on.

The Distribution will be 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.

After extraction, ZfD users associated with that Subscriber server will have access to the desktop applications that were distributed.

Rebuilding Desktop Application Distributions

Rebuilding a Desktop Application type of Distribution is done according to the established Build schedule. A rebuild can be triggered if the Revision number changes in either of two places:

- **Desktop Application Distribution object:** Any change to a Desktop Application Distribution object that causes its Revision number to change will trigger a rebuild of the Distribution. (Changing any of the object's properties will cause its Revision number to change.)

  The rebuild will include only the files that have changed since the last time the Distribution was built.

- **Application object:** Any change to an Application object included in a Desktop Application Distribution that causes the Application object's Revision number to change will trigger a rebuild of the Distribution. (Changing any of the Application object's properties will cause its Revision number to change.)

  The rebuild will include only the files that have changed since the last time the Distribution was built.

IMPORTANT: Simply adding or updating files to the file system will not alter the Revision number of the Application object or the Desktop Application Distribution object. Therefore, no rebuild will be triggered. However, added or updated files for an Application object will be included in the next rebuild when it is triggered.

Linking Site Lists

You can provide convenience for users who travel from site to site where you have replicated ZfD application objects using a TED distribution. Site lists can only be linked within the tree where the source application's object resides.

Review the following sections:

- "Distributions Tab for Application Objects" on page 691
- "Understanding Linked Site Lists" on page 691
- "Linking the Sites" on page 692

Distributions Tab for Application Objects

The ZfD properties page for an application object will have a Distributions tab once Policy and Distribution Services has been installed. There you can link up sites where the same application has been distributed.

The Distributions tab on this property page contains the following information:

- **Source Application:** Lists the name of the application object that was used to create the current application object; otherwise, the field is blank.
- **Replicated Applications:** Lists the ZfD application objects that have been replicated from the current application object using a Desktop Application Distribution. The listing is in a random order. The Link Up Site Lists button is used to link all of the listed application objects with each other.
- **Associated Distributions:** Displays the Desktop Application Distributions associated with the current application object.

Understanding Linked Site Lists

By linking Site Lists, you allow users who travel from site to site to have similar local access to their applications.

For example:

1. You create the App1 application object at SiteA.
2. You replicate App1 to SiteB and SiteC.
3. You click Link Up Site Lists on the Distributions tab in the App1 application object's properties.

4. Users who had access to App1 at SiteA will also have the same local access to App1 when they travel to SiteB or SiteC.

To further illustrate the concept of how site lists are linked, assume the following:

Original Application/Site	Replicated Applications/Sites
App1.SiteA	App1.SiteB
	App1.SiteC

Once you link the site lists, you have the following relationships:

Application/Site	Linked Applications/Sites
App1.SiteA	App1.SiteB App1.SiteC
App1.SiteB	App1.SiteA App1.SiteC
App1.SiteC	App1.SiteA App1.SiteB

Therefore, no matter which site the user is visiting, the application will be available in the same way that the user is used to.

Linking the Sites

To link up Site Lists,

1. In ConsoleOne, right-click an application object > click Properties > click the Distributions tab.
2. Click the Link Up Site Lists button.

   All replicated application objects listed in the Replicated Applications box are linked.

   You can unlink individual items using the Unlink button on the application object's Fault Tolerance > Links tab.

   The method for linking Site Lists described in these steps is faster than using the Link option on the Fault Tolerance > Links tab. The Link Up

Site Lists button links up all site lists related to the current application, where the Link option requires browsing for application objects to link them one at a time.

3. Click OK when finished reviewing the linked site list.

Any time you replicate an application to another site using the Desktop Application type of Distribution, you can repeat this procedure to update the linked sites.

21 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 695
- "Distribution Security Using Encryption" on page 702
- "Security for Inter-Server Communication Across Non-Secured Connections" on page 707

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. Thereafter, the certificate is stored in the .KEYSTORE file.

For information on security certificates for encrypted Distributions, see "Distribution Security Using Encryption" on page 702.

- **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 696
- "Understanding Certificate Usage in Policy and Distribution Services" on page 696
- "Important Points about Certificates" on page 698
- "Certificate File Locations" on page 698
- "Handling Invalid Certificates" on page 699
- "Certificate and Private Key Directories" on page 700
- "Creating Security Certificates for Non-Encrypted Distributions" on page 701
- "Manually Copying Certificates for Non-Encrypted Distributions" on page 702

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.

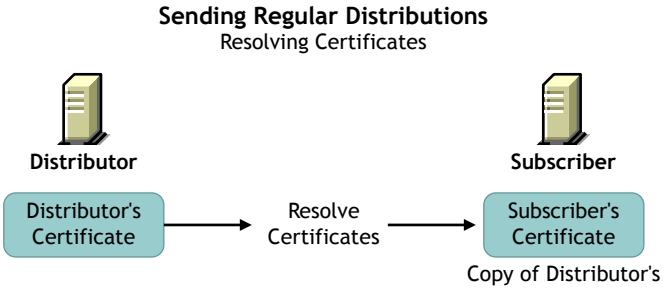
Understanding Certificate Usage in Policy and Distribution Services

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.

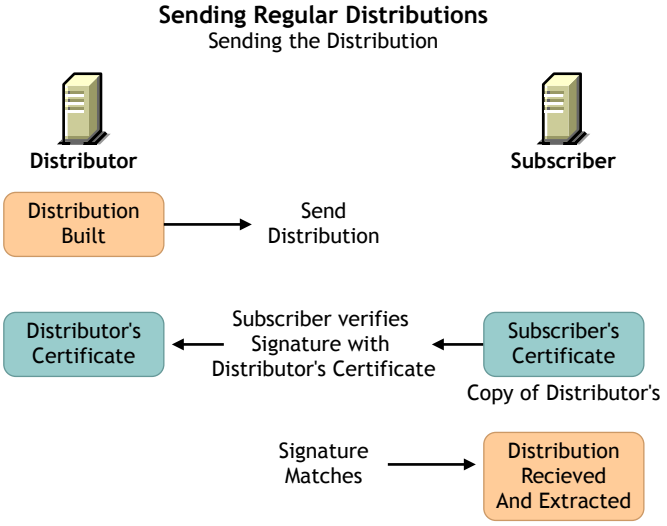
All Subscribers must receive a valid security certificate from each Distributor that sends Distributions to them. Without a matching certificate, a Subscriber cannot receive Distributions from the Distributor.

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.

After certificates have been resolved, the following illustrates how the Subscriber uses the certificate to ensure it is receiving a valid Distribution:



Important Points about Certificates

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 a certificate in its security directory 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.
- A certificate can be passed from a Distributor to a Subscriber under two circumstances:
  - When a Subscriber subscribes to a Channel and you click OK to apply the changes.
  - When a Distribution is listed in a Channel and you click OK to apply the changes.
  - When you resolve certificates from the menu option for the Distributor object.

  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 form the Subscriber's security subdirectory. Then the Distributor must send the new certificates to those Subscribers.

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`.

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 more information on encryption certificates, see "Distribution Security Using Encryption" on page 702.

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 699
- "Subscriber DNS Name or IP Address Is Changed" on page 700

Distributor DNS Name or IP Address Is Changed

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. To re-create and resolve the Distributor's certificate, do the following:

1. On the Distributor server, shut down the Distributor Agent.

   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 UNIX and Stopping the Policy and Distribution Services Agents on UNIX in Installing Policy and Distribution Services on UNIX 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. Restart the Distributor Agent.

   A new certificate and .KEYSTORE file will be automatically created for the Distributor.

4. To resolve certificates: 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.

Subscriber DNS Name or IP Address Is Changed

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 for each Distributor that sends it encrypted Distributions.

Subscribers will continue to receive non-encrypted Distributions, even if the DNS name or IP address is changed.

To reproduce valid encryption certificates for the Subscriber, follow the instructions under "Distribution Security Using Encryption" on page 702.

Certificate and Private Key Directories

Certificates and private keys for Policy and Distribution Services are stored in the following locations:

- 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

Security certificates are not normally encrypted. The creation process for encryption certificates is different. For more information, see "Distribution Security Using Encryption" on page 702.

To create a certificate on a Distributor and copy it to its associated Subscribers:

1. On the server where a Distributor is installed, run its Java* process (use TED.NCF on a NetWare server, or restart the TED Distribution service on a Windows server).

   This process creates the certificate and writes it into eDirectory.

2. Copy the certificate to each Subscriber using one of the following methods:

   - 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.

     This method is the easiest when there are many Subscribers receiving Distributions from one Distributor.

     Non-encrypted security certificates only need to be sent between a Distributor and Subscriber pair once. Thereafter, they will each use that certificate to verify Distributions.

     IMPORTANT: Distributors cannot copy certificates to Linux* and Solaris* servers, or to any server where certain DNS configurations exist. Therefore, it would be best to have drives mapped on the workstation where you are using ConsoleOne to resolve certificates so that when prompted you can point to the correct destinations for the certificates.

- Manually copy the Distributor's certificate from the ZENWORKS\PDS\TED\SECURITY\PRIVATE directory to each Subscriber server's ZENWORKS\PDS\TED\SECURITY directory.
- Right-click a Subscriber object > click Resolve Certificates > repeat for each Subscriber object.

3 Because each Distributor creates its own security certificate, repeat Step 1 and Step 2 for each Distributor object in the tree.

Where Subscribers receive Distributions from multiple Distributors, the Subscribers will have one certificate from each Distributor.

Manually Copying Certificates for Non-Encrypted Distributions

To manually copy certificates to Subscribers using ConsoleOne®, do any of the following:

- Right-click a Distributor, Subscriber, or External Subscriber object > click Resolve Certificates.
- Click File > Resolve Certificates.
- Click OK on a property page > click Yes when prompted whether to copy a certificate for that object.

  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

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.

Distribution encryption is not available for Linux or Solaris servers.

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.

Thereafter, the Distribution will be sent as an encrypted Distribution.

To understand Distribution encryption, review the following:

- “Creating and Copying Encryption Certificates” on page 703
- “Sending an Encrypted Distribution” on page 706
- “Extracting an Encrypted Distribution” on page 707

Creating and Copying Encryption Certificates

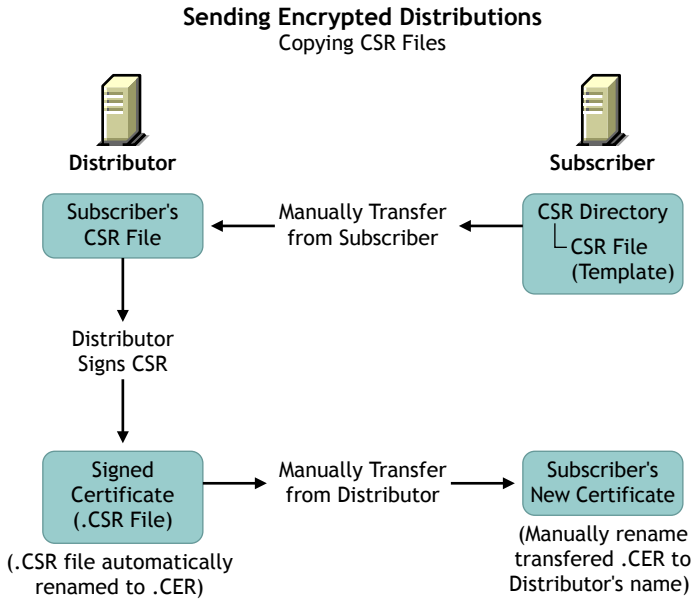
RSA PKIs provide the security process used for encrypted TED Distributions.

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.

To use encryption certificates with Subscribers, you must have previously resolved certificates and sent an non-encrypted Distribution to each Subscriber.

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.
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
```

HINT: 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:

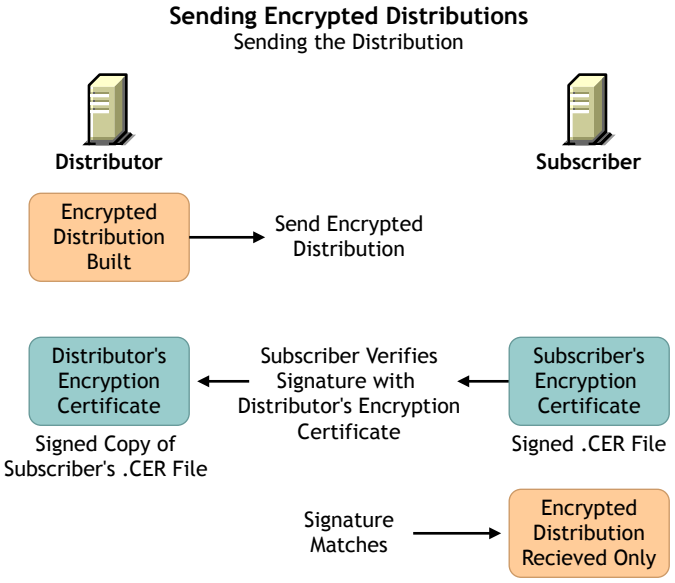
a. The encryption .CER file will be overwritten on each Subscriber with a non-encryption .CER file.

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

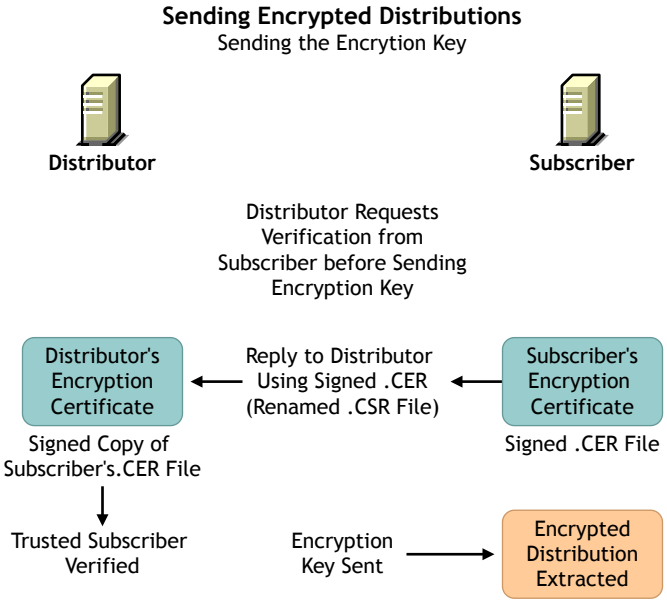
After an encryption certificate has been established on a Subscriber server, the following illustrates the process for sending encrypted Distributions:



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:



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 708
- "Security Certificates" on page 709
- "Using SSL for the ZenCSServlet" on page 710
- "Format of the Password File" on page 710

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.

22 Scheduling

This information is used in setting up each of the three Novell® ZENworks® for Servers (ZfS) Policy and Distribution Services components.

When you schedule a server policy or a Tiered Electronic Distribution (TED) Distribution, you can select the type of schedule.

Review the following sections:

- "Scheduling Server Policies" on page 713
- "Scheduling TED Objects" on page 714
- "Scheduling Issues" on page 716
- "Schedule Types" on page 722
- "Using Intervals and Repeating Actions in Schedule Types" on page 727

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 more information, see *Server Policies are not being enforced in the correct order* in the *Troubleshooting and Error Messages* guide.

Scheduling 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, 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 714
- "TED Object Schedules" on page 714

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 you manually set a schedule for a specific TED object. In other words, modified schedules for TED objects will override the Tiered Electronic Distribution policy schedule.

This is done by unchecking the Use Policy check box and setting a schedule. By default, when a schedule is set in the Tiered Electronic Distribution policy, the Use Policy check boxes are checked for the Distributor and Subscriber objects that have not had their schedules modified.

TED Object Schedules

The End Times for the following schedules, combined with the Start Time, specify the time window for starting the schedule's action:

- "Distributor Object's Refresh Schedule" on page 714
- "Distribution Object's Build Schedule" on page 715
- "Channel Object's Send Schedule" on page 715
- "Subscriber Object's Extract Schedule" on page 716

Distributor Object's Refresh 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 rebuild a Distribution when it discovers that there are configuration changes within eDirectory.

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, 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 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 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 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.

Distribution Object's Build Schedule

Determines when a Distributor will be requested to build the individual pieces that comprise the Distribution.

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 this schedule.

Channel Object's Send Schedule

Provides a window of time for when a Distributor can send its Distributions to the Subscribers.

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 the first tier Subscribers. Therefore, the second and lower tiers of Subscribers in the Distributor's routing hierarchy do not adhere to the Send schedule.

The Channel's End Time is enforced for this schedule. The Distributor will stop sending a Distribution when the Channel schedule ends and will start sending the Distribution where it left off when the Channel 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.

Note that this End Time enforcement only applies to a Distributor sending a Distribution to a first-tier Subscriber, not to parent Subscribers sending to other Subscribers. Parent Subscribers who are passing a Distribution on will ignore the Channel's End Time and complete sending the Distribution.

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.

Subscriber Object's Extract Schedule

Determines when a Subscriber can start to 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 Issues

The following explain various scheduling issues:

- "Scheduling Differences Between Policies and TED" on page 717
- "Scheduling Conflicts with Other Software" on page 717
- "Randomly Dispatch Option Issues" on page 717
- "Distributor Scheduling Issues" on page 719
- "TED Object Scheduling Issues" on page 720
- "Inactivating Distributions and Channels" on page 722

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.

The following describe the issues for the Randomly Dispatch option:

- "Using the Randomly Dispatch Option in a Distributor's Refresh Schedule" on page 718
- "Using the Randomly Dispatch Option in a Distribution's Build Schedule" on page 718

- "Using the Randomly Dispatch Option in a Channel's Send Schedule" on page 719
- "Using the Randomly Dispatch Option in a Subscriber's Extract Schedule" on page 719

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 three minutes.

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 three minutes.

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 720).

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) 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.

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.

Schedule Type	Description
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 was refreshed. For example, if you set the time to one hour, and refresh the policy, it will run in one hour. 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 490).

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 is refreshed:

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.

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 728
- “Repeating Actions” on page 728

Using Intervals with Distributors

For any schedule type the 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 run 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 16, "Novell iManager," on page 481.

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 (still in the queue), 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.

23 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 731
- "Types of Variables" on page 732
- "Resolution of Variable Names" on page 734
- "Nested Variables" on page 735
- "Creating a Variable" on page 735
- "Using Variables to Control File Extraction" on page 738

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 732
- "Where Variables Can Be Used" on page 732

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 733
- "User-Defined Variables" on page 734

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\PDS\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.

Variable	Description
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 735
- "Creating Variables for a Specific Subscriber" on page 736
- "Creating Variables for a Software Package" on page 737

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 the Distribution" on page 560.

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.

24 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 741
- "Installing, Setting Up, and Connecting To the ZENworks Database" on page 748
- "Creating a ZENworks Database Object" on page 755
- "Purging the Database" on page 756
- "Database Contents" on page 756

Understanding the ZENworks Database

The following sections provide an understanding of the ZENworks database:

- "The Database File" on page 742
- "Database File Location" on page 742
- "The Database Object" on page 742
- "Running the Database" on page 390
- "Database Caching" on page 390
- "Database Information" on page 742
- "Coexisting Databases" on page 744

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 Wizard. 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 more information on purging, see "Purging the Database" on page 756.

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: ◆ 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 25, "Reporting," on page 771.

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 Wizard 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 Wizard. 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 745
- "Multiple Databases" on page 746

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

Server Policies Reports	TED Reports
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 746.

For more information on reporting, see Chapter 25, "Reporting," on page 771.

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 746
- "Distributor Object Contexts and Multiple Databases" on page 747
- "Determining Whether You Need Multiple Databases" on page 748

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 type of 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 Wizard 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 749
- "Configuring the ZENworks Database Policy" on page 751
- "Connecting to the Database" on page 753

Installing the Database

To install the ZENworks database:

1. On the workstation, insert the *ZENworks for Servers* product 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 Distributions Services option.
3. Review the License Agreement > click Accept if you agree > click Next.

   or

   Click Decline > click Next to exit the Installation Wizard.
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 Wizard now copies files and installs the database objects.

WARNING: If you click Cancel, none of the work you did in the Installation Wizard 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 751).

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 751
- "For the Policy/Package Agent" on page 752

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 Close > 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 Close > 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 Close > 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 633.

    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	1. Open the Control Panel. 2. Click Services (in Windows 2000, Services is under Admin Tools). 3. Click Novell ZfS Policies > click Start.

Server Platform	Agent Startup Method
NetWare	**SYS:\ZENWORKS\PDS\SMANAGER\ZFS.NCF**
Solaris* or Linux*	**/usr/ZENworks/pds/smanager/ZFSSRV.sh**

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	1. Open the Control Panel. 2. Click Services (in Windows 2000, Services is under Admin Tools). 3. Click Novell ZfS Distribution > click Start.
NetWare	**SYS:\ZENWORKS\PDS\SMANAGER\TED.NCF**
Solaris or Linux	**/usr/ZENworks/pds/smanager/TEDSRV.sh**

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 630.
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.

Database Contents

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 25, “Reporting,” on page 771) or iManager (Chapter 16, “Novell iManager,” on page 481) for determining the status of Distributions or policies.

The database file (ZFSLOG.DB) contains the following information:

- “Server Policies Database Contents” on page 757
- “TED Database Contents” on page 764

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 759.
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 760.
ACTIONCODE	integer		The action being performed. For definitions, see “Valid Entries for ACTIONCODE” on page 760.

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 761.
CODE	integer		Code representing the results of the action. For definitions, see "Valid Entries for CODE" on page 761.
ACTIONCODE	integer		Code representing the action being performed. For definitions, see "Valid Entries for ACTIONCODE" on page 762.
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 763.
CODE	integer		Code representing the results of the action. For definitions, see “Valid Entries for CODE” on page 763.
ACTIONCODE	integer		Code representing the action being performed. For definitions, see “Valid Entries for ACTIONCODE” on page 763.

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 765
- “TAB_CHANNEL” on page 765
- “TAB_DISTRIBUTION” on page 766
- “TAB_DIST_VERSION” on page 766
- “TAB_DIST_ACTION” on page 767

- "TAB_CHANNEL_DISTRIBUTION" on page 768
- "Foreign Keys" on page 769

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)		not null	TED object DN.
TYPE	char		not null	"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			Use
ID	numeric(12,0)	identity	not null	Unique number automatically assigned.
DIST_VERSION_ID	numeric(10,0)		not null	Links to the TAB_DIST_VERSION table.
NODE_ID	numeric(8,0)		not null	Links to the TAB_NODE table for the node performing the following tasks: Create Send Receive Extract Post process
TIMESTAMP	datetime		not null	Time stamp when the action was logged into the database.
STAGE	char		not null	"C"=Create "S"=Send "R"=Receive "E"=Extract "P"=Post process
STATUS	char		not null	"S"=Success "F"=Failure "P"=In process
STATUS_TIMESTAMP	datetime		not null	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 768.
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;";

25 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 "Database Contents" on page 756.

Review the following:

- "Storing Report Information" on page 772
- "Reporting Scope for TED Objects" on page 772
- "Reporting on the Successes and Failures of Distributions" on page 772
- "Generating Reports" on page 773
- "Report Descriptions" on page 774

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
   Failed Policies
   Packages
   Server Down Process Policy
   Successful Policies
   Unenforceable Policies

4. Select the reporting criteria.

   If you need more detail on reporting criteria or content, see "Report Descriptions" on page 774.

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 774
- "Server Policy Reports" on page 776

TED Reports

There are four predefined TED reports:

- "Distribution Detail Report" on page 774
- "Revision History Report" on page 775
- "Revision History Failure Report" on page 775
- "Subscriber Detail Report" on page 776

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 777
- "Server Down Process Report" on page 777
- "Failed Policies Report" on page 777
- "Packages Report" on page 778
- "Successful Policies Report" on page 778
- "Unenforceable Policies Report" on page 779

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.PRV.NOVELL is sorted NOVELL, PRV, 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.PRV.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.PRV.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.PRV.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.

26 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 ZENworks® for Servers (ZfS) 3. The information will help you to keep current on updates to the documentation.

The 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.

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.

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 in the Legal Notices section immediately following the title page.

The documentation was updated on the following dates:

- "May 17, 2002" on page 782
- "June 6, 2002" on page 784
- "September 27, 2002" on page 785
- "December 19, 2002" on page 786

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 446	NICI must be installed on Windows servers where encrypted Distributions are either built or extracted. A new planning section was added.
"Installing NICI on Windows Servers" on page 458	Added this new configuration section for NICI.
"Configuration Planning Worksheet" on page 470	Inserted two new worksheet items at the beginning for installing NICI, causing all other worksheet items to be renumbered.
"Understanding External Subscribers" on page 514	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 *ZENworks for Servers Companion* CD under ZFS\TEDPOL\SFILES\TED.
"Multiple Databases" on page 746	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 453	Clarified database installation: only one can be installed per run of the Installation Wizard. 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 580	Added this new section to document why and how to refresh the Distributor.
“Tiered Electronic Distribution Policy” on page 540	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 549. Step 5 under Value in “Tiered Electronic Distribution” on page 613. Step 4 in “Creating Default Variables for All Subscribers” on page 735. Step 4 in “Creating Variables for a Specific Subscriber” on page 736. Step 4 in “Using Variables to Control File Extraction” on page 738.

Automating Server Software Installations and Updates

The following changes were made in this section:

Location	Change
“Converting Older Server Software Packages to ZfS 3” on page 679	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 707	Moved all of the installation information inter-server communications security into the *Installation* guide. For the new location of the moved information, see Installing Additional Security for Non-secured Connections under Installing ZENworks for Servers in the *Installation* guide.

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 440	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 699	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.

September 27, 2002

Updates were made to the following sections. The changes are explained below.

- Policy and Distribution Services
- Configuring Policy and Distribution Services
- Tiered Electronic Distribution
- 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 421	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.
"Appendixes" on page 1087	Moved the information in the References section to the Appendixes section.

Configuring Policy and Distribution Services

The following changes were made in this section:

Location	Change
Chapter 15, "Configuring Policy and Distribution Services," on page 423	Reorganized and rewrote this entire section to correspond to the changes in Installing Policy and Distribution Services on NetWare and Windows Servers in the *Installation* guide.

Tiered Electronic Distribution

The following changes were made in this section:

Location	Change
"Creating Security Certificates" on page 574	Added a new first bullet under Step 2 on page 575 that explains how to resolve certificates using the Distributor object.

Novell iManager

The following changes were made in this section:

Location	Change
"Managing the Policy/ Package Agent from the Remote Web Console" on page 491	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 493	You can now create the Desktop Application type of Distribution in Novell i Manager.

Security in Policy and Distribution Services

The following changes were made in this section:

Location	Change
"Handling Invalid Certificates" on page 699	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.

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

Configuring Policy and Distribution Services

The following changes were made in this section:

Location	Change
"Desktop Application" on page 431	Added a new second paragraph concerning inter-tree distributions of Application objects: 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.
"Desktop Application" on page 431	Added the following new fourth paragraph: 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.
"Desktop Application" on page 431	Added the following item in the worksheet entry table. 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.
"Software Package" on page 434	Added a new note concerning the order of .CPK file installations when you have multiple software packages in one Distribution: **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.

Tiered Electronic Distribution

The following changes were made in this section:

Location	Change
"Maximum Revisions" on page 558	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 528	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 644	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 650	Added this new section to explain management issues related to software package files.
"Setting Up Multiple-Workstation Management for Server Software Packages" on page 658	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 690	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 734	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
"Database Contents" on page 756	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 25, "Reporting," on page 771) or iManager (Chapter 16, "Novell iManager," on page 481) for determining the status of Distributions or policies.

Using Server Software Packages to Delete Directories on Servers

The following changes were made in this section, which is contained in "Appendixes" on page 1087:

Location	Change
Appendix E, "Using Server Software Packages to Delete Directories on Servers," on page 1113	Added this new section that explains how to use software packages to delete directories on network servers.