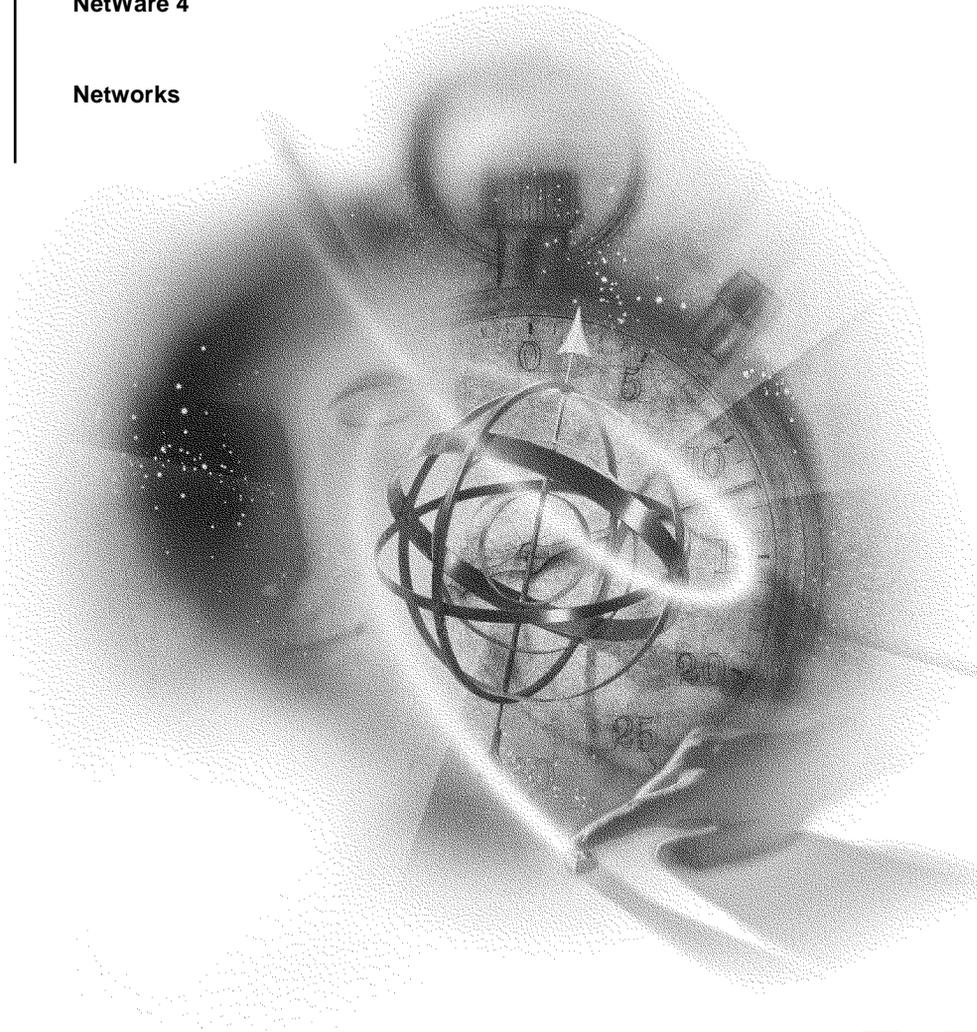


VERSION 4.2

Guide to

NetWare 4

Networks



NetWare[®] 4
NETWORK SOFTWARE

Novell[®]

disclaimer

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

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

export notice

This product may require export authorization from the U.S. Department of Commerce prior to exporting from the U.S. or Canada.

trademarks

Novell and NetWare are registered trademarks of Novell, Inc. in the United States and other countries.

A complete list of trademarks and their respective owners appears in "Trademarks" on page 267.

Copyright © 1993-1999 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

U.S. Patent Nos. 5,157,663; 5,349,642; 5,455,932, 5,553,139; 5,553,143; 5,594,863; 5,608,903; 5,633,931; 5,652,854; 5,671,414; 5,677,851; 5,692,129. U.S. and Foreign Patents Pending.

**Novell, Inc.
122 East 1700 South
Provo, UT 84606
U.S.A.**

www.novell.com

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

**Guide to NetWare 4 Networks
December 1998
104-000040-001**

Contents

How to Use This Manual

- Introduction ix
- General Process and Procedures x
- Overview of This Guide xii

1 Organizing and Training the Project Team

- Defining Your Team Organization. 1
 - Assessing the Necessary Skills 2
 - Identifying Member Roles 2
- Determining Team Training Needs 10
 - Identifying Training Topics 11
 - Identifying Training Opportunities 13
- Where to Go from Here 14

2 Gathering Information and Defining the Project Scope

- Determining What Information Is Needed. 15
 - MIS Manager 16
 - NDS Expert 16
 - NetWare Server Specialist 17
 - Workstation Specialist 17
 - Application Specialist 18
 - Printing Specialist 18
 - Connectivity Specialist 19
 - Testing Lab Coordinator. 19
 - Education and Training Coordinator. 20
- Defining the Project Design Scope 20
 - Identifying Critical Factors. 20
 - Creating a Design Phase Schedule 22
 - Creating New Documentation 23
- Where to Go from Here 23

3 Designing the Directory Tree Structure

Introduction	25
Object-Oriented Database	26
Object Classes	27
Directory Rules and Definitions	27
Directory Tree Structure	28
Directory Tree Design	28
Objectives	30
Prerequisites	30
Creating a Naming Standards Document	30
Using a Naming Standards Template	31
Identifying Naming Considerations	33
Planning the Directory Tree Structure	37
Designing Upper Layers	38
Designing Lower Layers	46
Reviewing the Directory Tree Structure	52
Planning Placement of Network Resources	52
Identifying Leaf Object Types	52
Determining Leaf Object Placement	60
Summary	64
Evaluation	64
Defaults.	65
Where to Go from Here	66

4 Determining a Partition and Replication Strategy

Introduction	67
Objectives	68
Prerequisites	68
Determining Partition Requirements	69
Forming Partition Boundaries	70
Considering Partition Characteristics.	70
Planning the Partition Layout.	72
Determining an Efficient Replica Placement Strategy	75
Considering Replica Characteristics	75
Planning Replica Placement	79
Creating a Replication Matrix.	84
Summary	84
Evaluation	85
Defaults.	86
Where to Go from Here	87

5 Planning the Time Synchronization Strategy

Introduction	90
Objectives	91
Prerequisites	91
Determining an Efficient Time Synchronization Method	91
Using Internal Time Synchronization Methods	92
Using External Time Synchronization Methods	94
Using a Combination of Time Synchronization Methods	94
Identifying an Efficient Time Source and Configuration	95
Identifying an Efficient Time Source	95
Determining an Efficient Configuration	96
Identifying the Communication Format	101
Summary	103
Evaluation	104
Defaults	104
Where to Go from Here	105

6 Creating an Accessibility Plan

Introduction	107
Objectives	108
Prerequisites	108
Understanding How Network Resources Are Accessed	109
Identifying Objects by Name	109
Identifying Directory Objects by Location	112
Using Access-Related Objects	113
Determining Access Needs	115
Identifying Network Connection Types	116
Identifying Novell Client Types	117
Identifying User Types	117
Identifying Global and Shared Resources	118
Identifying Bindery Services Needs	119
Determining an Efficient Access Control Method	123
Authentication	123
NDS and File System Security	124
Login and Profile Scripting	136
Administrative Objects	141
Summary	142
Evaluation	143
Defaults	143
Where to Go from Here	148

7 Designing a Data Protection Plan

Introduction	149
Objectives	150
Prerequisites	150
Establishing a Redundant Hardware System	150
Ensuring Adequate Partitioning and Replication	151
Developing a Backup and Restore Strategy	152
Third-Party Solutions	152
Compatibility	153
Determining Backup Administration Strategy	153
Considering Network-Related Issues	154
Data Protection Guidelines	155
Creating a Disaster Prevention and Recovery Plan	157
Summary	158
Where to Go from Here	158

8 Designing an Application Management Strategy

Introduction	159
Objectives	160
Prerequisites	161
Ensuring NetWare Compatibility	161
Novell's Yes Program	161
Ensuring Applications Are Designed for Multiple-User Access	162
Using the Application Compatibility Template	162
Creating Efficient and Intuitive Application Directories	162
Creating the Application Directory Structure	162
Providing Adequate Load Balancing	163
Protecting Application Directories	164
Identifying Efficient Application Management Tools	165
Using NetWare Application Management Tools	165
Identifying Efficient Licensing and Metering Tools	168
Metering Applications	168
Licensing Applications	168
Application Management Guidelines	170
Summary	171
Where to Go from Here	171

9 Developing a Migration Strategy

Introduction	174
Objectives	174
Prerequisites	175
Determining a Client Migration Method	175
Migrating Client Software before Installing NetWare 4	175

Determining a Server Migration Method	181
Identifying Upgrade Methods	181
The Novell Upgrade Wizard	182
Identifying Other Upgrade Options	183
Maintaining Backwards Compatibility	184
Maintaining Bindery Services in a NetWare 4 Environment.	184
Maintaining NetWare 3 Servers Using NetSync.	189
Maintaining a Mixed NetWare 4 Environment.	190
Setting Up a Lab	191
Using NetWare 4.2 Utilities	191
Analyzing Hardware and Hardware Driver Compatibility	192
Establishing a Pilot System	193
Summary	194
Evaluation	194
Where to Go from Here	195

10 Creating an Implementation Schedule

Introduction	197
Objectives	198
Prerequisites	198
Identifying Tasks and Assignments	198
Creating a Task List	198
Assigning Task Responsibilities	199
Determining an Efficient Implementation Schedule	199
Scheduling Tasks	199
Tracking Progress	200
Creating an Implementation Schedule Draft	200
Summary	200
Where to Go from Here	201

11 Implementing NetWare 4 Services

Introduction	203
Objectives	203
Prerequisites	204
Completing General Tasks	204
Implementing NDS on Various Network Types	208
Single-Site Network	208
Multiple-Site Network	210
Multiple-Campus Network	214
Summary	219

A NDS Object Classes and Properties

Overview	221
NDS Object Classes and Their Functions	222
NDS Object Classes and Their Properties	225

B Referencing and Using Leaf Objects

Overview	239
Application Leaf Object	240
Auditing Leaf Object	240
Informational Leaf Object	241
Messaging-Related Leaf Objects	241
Miscellaneous Leaf Object	243
NetWare Licensing Services Leaf Object	244
Printer-Related Leaf Objects	245
Server-Related Leaf Objects	246
User-Related Leaf Objects	247

C Template Examples

Application Compatibility	252
Implementation Schedule	253
Name Standards	254
NetWare 4 Server Worksheet	255
Replica Placement Worksheet	257
Server Migration	258
Workstation Configuration Worksheet	259

D Supplemental Information

Resources	261
Online Help	261
Additional Help Resources	262

Trademarks

Novell Trademarks	267
Third-Party Trademarks	273

How to Use This Manual

Introduction

All network installations, regardless of size or complexity, benefit from sound planning and efficient implementation. The goal of this guide is to give you practical information regarding processes, guidelines, and rationale for planning and implementing a NetWare® 4™ network.

Much of the information included in this guide originated from extensive research and testing done by consultants in the Novell® Consulting Services group. The processes and rationale in this guide are derived from case-proven operations performed by these consultants.

As with any implementation of a new product release, some transitioning is required. Because of NetWare 4's wide range of new technologies, features, and support, the transition may appear greater than it really is.

By following the processes and guidelines in this guide, you will discover the best approach and strategy for transitioning your network environment to NetWare 4 and identify necessary tasks and scale your NetWare 4 project to meet the needs of your organization.

The processes and guidelines that are provided can be easily tailored to your particular project and network infrastructure.

Once you complete your NetWare 4 implementation, you will see the numerous benefits of using NetWare 4 in your network environment.

Some of the obvious benefits are the following:

- ◆ Sizing your network into a single view of the entire network for simpler access, use, and administration.
- ◆ Protecting your existing investments by more than doubling disk capacity and adding numerous network services without added hardware expense.

- ◆ Supporting previous NetWare installations by providing full backward compatibility with NetWare 3™.
- ◆ Providing a high level of scalability for managing growth and changes in your network.

General Process and Procedures

The general process and procedures for designing and implementing NetWare 4 can be divided into three phases. These phases are the following:

- ◆ Project approach
- ◆ Design
- ◆ Implementation

Each phase contains a series of procedures you may or may not need to perform, depending on your particular network environment.

Phase	Procedure (Section)	Rationale
Project Approach	“Organizing and Training the Project Team”	Helps the project team set realistic expectations so that the design and implementation will proceed smoothly. Organizing the project in this way also gives members of the team an idea of their roles throughout the process.
	“Gathering Information and Defining the Project Scope”	Helps the project team identify specific resources, organizational and workgroup access, and workflow processes within your organization.
Design	“Designing the Directory Tree Structure”	Helps you set up <i>Novell Directory Services</i> (NDS) so that it works efficiently, is easy for network users to employ, and is easy for administrators to manage. It also lays a foundation for completing of the next procedure, designing partitions and replicas.

Phase	Procedure (Section)	Rationale
Design (<i>cont'd.</i>)	**"Determining a Partition and Replication Strategy"	Helps provide the scalability, fault tolerance, and accessibility of the Directory across the network.
	**"Planning the Time Synchronization Strategy"	Helps provide accurate, synchronized time to all servers on the network. The servers can then apply time stamps to NDS events and to other services such as messaging applications and file systems.
	"Creating an Accessibility Plan"	Makes it easier to access network resources and services.
	**"Designing a Data Protection Plan"	Helps you avoid data loss or disasters.
	**"Designing an Application Management Strategy"	Helps you effectively set up your application for easier access, load balance, and fault tolerance.
Implementation	**"Developing a Migration Strategy"	Makes the actual upgrades and migrations go smoothly. Setting up a lab helps you avoid the problems that administrators could otherwise encounter when trying to implement NetWare 4.
	"Creating an Implementation Schedule"	Helps prepare the various implementation teams, eliminates confusion, provides efficient execution of tasks, and communicates migration tasks.
	"Implementing NetWare 4 Services"	Allows for a trouble free and timely implementation of NetWare 4.
* conditional procedure		

Identifying Conditional Procedures

Some procedures are labeled as *conditional*. When beginning a specific procedure, you should first identify if the procedure is conditional or not. If so, determine if you need to perform that procedure for your network.

For example, a procedure for designing time synchronization requires that your network have a WAN link or over thirty servers in the network.

The discussion for each conditional procedure begins with a list of requirements to help you in decide if you need to perform a specific procedure. See Procedure Requirements at the beginning of each section for information.

Overview of This Guide

This guide is a central reference point for information about NetWare 4 design and implementation. After reading this guide, you should know how to design and implement a NetWare 4 network.

Pointers are provided for accessing only the pertinent information for your particular network type. Administrators of simple networks are guided to information that addresses their particular needs. Administrators of complex networks, or those that have WAN link considerations, can access critical information useful during their planning process.

This guide contains processes, rationale, and examples of current implementations and applications of NetWare 4. For more detail, references are provided to the documentation included with NetWare 4.

Note: In Novell documentation, an asterisk denotes a trademarked name belonging to a third-party company. Novell trademarks are denoted with specific trademark symbols, such as TM.

This guide is not a resource for conceptual information about various features of NetWare 4. It also does not address issues about complex and specific configuration of NetWare 4. Those issues are discussed in Novell Application NotesTM or can be obtained from Novell Technical Support documents.

Organizing and Training the Project Team

Organizing and training the NetWare® 4™ project team consists of defining the team organization, identifying roles, and educating team members about NetWare 4 concepts, features, and use. The team will manage, plan, and be responsible for all of the implementation tasks to be accomplished.

The following topics are discussed in this chapter:

- ◆ “Defining Your Team Organization” on page 1
- ◆ “Determining Team Training Needs” on page 10

Defining Your Team Organization

The size of your organization and complexity of your NetWare 4 implementation determines the framework of your project team. Some teams may have one consultant designing and implementing a single server and ten workstations, where other teams may have numerous people assigned to different specialities across the organization, designing and implementing a multisite network with hundreds of servers and thousands of workstations.

Organizing the project team is critical to the efficient design and implementation of NetWare 4. The team ensures that implementing NetWare 4 is as smooth as possible.

Before you organize your team, you should:

- ◆ Assess the skills needed to design and implement NetWare 4.
- ◆ Identify the necessary member roles within the team and who will perform those roles.

Assessing the Necessary Skills

The project manager chooses the team members according to responsibilities and skill sets. Organizations with preexisting NetWare networks can build on the existing skill sets developed from working with previous versions of NetWare.

Organizations setting up NetWare 4 as a new network need to identify people with basic networking and communications skills in addition to expertise in setting up and configuring NetWare networks.

Identifying Member Roles

It is not important how many people are on your project team, however, all expectations and concerns for each team member role should be represented by some team member.

Each team member will maintain a high level of expertise in areas of management, network administration, or user support service. Not all team members need to come from your particular organization. For example, it is typical that the wire layout and communication design is managed by an outside group.

Determining the Right Person for the Right Role

To determine the person best suited for a specific role, refer to the following list of questions:

- How is the Information Services (IS) department organized?
- Who manages client workstation setup and configuration?
- Who manages server setup and configuration?
- Who manages the physical network of LAN and WAN hardware and software?
- Who makes management decisions concerning networking hardware and software?
- If your network wire layout or protocols affect the implementation of NetWare 4, who is responsible for managing the network wire layout?

- Who provides training for management, administrators, and users?
- Who manages software upgrades?
- Who manages network resources such as printers, backup software and hardware, file storage, CD-ROM drives, etc.?
- Who tests and analyzes hardware and software for use on the network?

Identifying Role Responsibilities

When identifying the specific responsibilities of a project team role, you should understand the priorities, expectations, and concerns of each role. The following lists will help you identify the responsibilities of each role.

MIS Manager

This person is a manager or administrator in the group that manages design, implementation, and maintenance of your network. This person will commonly be project team lead throughout the implementation phase and possibly throughout the design phase.

- Priorities
 - ◆ Coordinating with the Novell® Directory Services™ (NDS™) expert to ensure an efficient transition
 - ◆ Acquiring the appropriate resources and funding to proceed with the design and implementation
 - ◆ Overseeing the design phase
 - ◆ Coordinating and supervising the implementation phase
- Expectations
 - ◆ Providing direction to the project
 - ◆ Conveying concerns to and from upper management and other departments
 - ◆ Managing costs
 - ◆ Organizing meetings

Concerns

- ◆ Designing an efficient network
- ◆ Managing the cost of implementation, operation, software licensing, etc.
- ◆ Evaluating software
- ◆ Creating a timeline for implementation
- ◆ Handling communication between departments and project team members
- ◆ Affecting user productivity
- ◆ Training administrators and users
- ◆ Developing a method and procedure for rollout
- ◆ Coordinating the pilot implementation
- ◆ Providing support after implementation

NDS Expert

This person has worked with NetWare 4 and NDS or has completed training relative to NetWare 4 and NDS. (This role could also be filled by a NetWare consultant.) The NDS expert would most likely be the team lead in the design process and possibly throughout the implementation process.

Priorities

- ◆ Creating a Directory tree design
- ◆ Designing NDS security
- ◆ Formulating partitioning
- ◆ Choosing project team members

Expectations

- ◆ Creating a solid design
- ◆ Ensuring that the design is thorough and meets all department needs

- ◆ Ensuring project team member participation and input
- ◆ Meeting timelines

Concerns

- ◆ Managing expectations of each department and of management
- ◆ Understanding NDS
- ◆ Coordinating login scripts with other concerned team members
- ◆ Coordinating with other groups
- ◆ Assigning someone to document the design

NetWare Server Specialist

A server specialist is someone who works daily with NetWare server administration.

Priorities

- ◆ Maintaining network performance levels
- ◆ Determining and planning pilot system
- ◆ Designing time synchronization
- ◆ Implementing upgrade and migration for the rollout

Expectations

- ◆ Helping with the lab
- ◆ Reducing administration
- ◆ Creating standards for protocols usage
- ◆ Reviewing and creating standard for frametype usage

Concerns

- ◆ Ensuring stability of the network
- ◆ Ensuring efficiency and performance

- ◆ Planning server placement in the Directory tree
- ◆ Calculating needed disk space for new and existing servers
- ◆ Maintaining and managing hardware
- ◆ Backing up file system and NDS data
- ◆ Investigating changes in server utilities
- ◆ Preparing for disaster recovery
- ◆ Ensuring backwards compatibility
- ◆ Ensuring that the server IPX™ address is registered with Novell, Inc.
- ◆ Removing and adding servers
- ◆ Identifying and setting the bindery context for users

Workstation Specialist

This person works daily with users and workstations. This person maintains login scripts and loads and upgrades network client software.

Priorities

- ◆ Upgrading workstations to the current Novell Client™ Software
- ◆ Automating the client software upgrade

Expectations

- ◆ Identifying differences between the Novell Client software, such as differences between the NetWare DOS Requester™ and Novell Client Software
- ◆ Creating an efficient implementation schedule for workstations
- ◆ Updating Novell Client software
- ◆ Identifying any problems with workstation hardware

Concerns

- ◆ Determining memory requirements to run Novell Client Software
- ◆ Identifying necessary changes to configuration files
- ◆ Automating the upgrade process
- ◆ Identifying performance issues relative to memory usage
- ◆ Determining and setting the name context
- ◆ Coordinating login scripts with NDS team members
- ◆ Testing compatibility
- ◆ Determining bindery services needs
- ◆ Determining mobile client issues

Application Specialist

This person maintains the application servers and software upgrades on client workstations and updates menu systems.

Priorities

- ◆ Migrating applications on application servers and client workstations
- ◆ Testing compatibility of applications

Expectations

- ◆ Ensuring stability of applications
- ◆ Providing user access to applications
- ◆ Updating menu systems or other access to applications
- ◆ Coordinating with NetWare Server Specialist

Concerns

- ◆ Determining which applications require support through bindery services
- ◆ Identifying any compatibility issues

- ◆ Ensuring that data is migrated properly
- ◆ Developing efficient processes for maintaining applications
- ◆ Coordinating login scripts with NDS team members

Printing Specialist

This person provides access to printers, determines printer locations, and upgrades printing software.

Priorities

- ◆ Providing users with access to printers
- ◆ Upgrading printer software and drivers

Expectations

- ◆ Creating a well-defined migration strategy
- ◆ Providing easy access to printing resources
- ◆ Providing efficient use of printing resources

Concerns

- ◆ Determining how to set up direct connect printers as separate network nodes
- ◆ Configuring and administering print services
- ◆ Identifying differences between current and previous print utilities
- ◆ Coordinating login scripts with team members

Connectivity Specialist

This person maintains the physical network, the network backbone, telecommunications, WAN design, and router placement.

Priorities

- ◆ Determining the effect of routing, protocols, telecommunications, and WAN structure on the Directory tree design
- ◆ Making decisions regarding single or multiple protocols on the network

- ❑ Expectations
 - ◆ Improving network traffic
 - ◆ Advising the team about routing, protocols, and WAN structure
- ❑ Concerns
 - ◆ Ensuring that an efficient balance exists between the network design and performance of the network over WAN links
 - ◆ Identifying LAN/WAN bandwidth issues
 - ◆ Identifying necessary protocols and applications used on the network
 - ◆ Setting up and maintaining single or multiple protocols

Testing Lab Coordinator

This person sets up and tests the networking software with current applications, runs diagnostics, and provides statistics on performance and network and application software stability.

- ❑ Priorities
 - ◆ Testing the compatibility between networking software and applications
 - ◆ Creating network performance benchmarks
- ❑ Expectations
 - ◆ Providing test results about network performance and software compatibility
 - ◆ Setting up the lab
- ❑ Concerns
 - ◆ Gathering all network information in order to simulate the network environment in the lab
 - ◆ Gathering current versions of application software
 - ◆ Obtaining resources for lab

Education and Training Coordinator

This person analyzes the skills required and creates or provides training to help administrators and users with network operating procedures.

- ☐ Priorities
 - ◆ Identifying and providing implementation and administration guidelines
 - ◆ Training the project team
 - ◆ Training network administrators
 - ◆ Training users

- ☐ Expectations
 - ◆ Working closely with project lead
 - ◆ Researching information that impacts users and administrators
 - ◆ Performing on-the-job-training

- ☐ Concerns
 - ◆ Budgeting for training
 - ◆ Determining the scope of training
 - ◆ Setting up the lab to train administrators
 - ◆ Scheduling training to meet project timelines
 - ◆ Getting team members up to speed before meetings

Determining Team Training Needs

The Education and Training Coordinator is responsible for providing the necessary training. Education and training can be provided in many formats, such as a lab environment for hands-on training or training courses through internal or external groups.

Identifying Training Topics

Use the following checklist of NetWare 4 concepts to determine the prerequisite training needs for your project team members. These concepts should be understood before you start your design process.

Basic NetWare 4 Concepts

- NetWare 4 terminology and concepts
- Novell Directory Services overview
 - ◆ Novell Directory Services objects and properties
 - ◆ Object organization
 - ◆ Directory tree design
 - ◆ Novell Directory Services partitions and replicas
 - ◆ Time synchronization
 - ◆ Bindery services

- Workstation
 - ◆ Novell Client for DOS and Windows* 3.1x
 - ◆ Novell Client for Windows 95/98
 - ◆ Novell Client for Windows NT*
 - ◆ Workstation configuration

See the Novell Client documentation for more information.

- NetWare 4 utilities
 - ◆ Administrative utilities
 - ◆ User utilities

See *Utilities Reference* and *Supervising the Network* for more information.

- ❑ Migrating to NetWare 4
 - ◆ New NetWare 4 installation
 - ◆ Migrating NetWare 3™ to NetWare 4
 - ◆ Upgrading bindery to Novell Directory Services

See *Installation and Upgrade* for more information.

New NetWare 4 Features

- ❑ File system
 - ◆ File compression
 - ◆ Suballocation blocks
 - ◆ Data migration to high capacity storage systems (HCSS)
- ❑ Auditing
 - ◆ AUDITCON utility
 - ◆ User auditor
- ❑ NetWare 4 Print Services
 - ◆ New features and requirements
 - ◆ Printing software and utilities
- ❑ Backup and restore
 - ◆ Hardware and software independence
 - ◆ Types of backup systems
 - ◆ Target Service Agent (TSA)

Advanced NetWare 4 Concepts

- ❑ Architecture
- ❑ NDS synchronization
- ❑ Internationalization

See *Supervising the Network and Concepts* for more information.

Identifying Training Opportunities

There are several types and sources of NetWare training available.

Novell Education

Instructor-led courses are available through Novell Authorized Education CentersSM (NAECSM) and Novell Academic Education Partners^{CM} (NAEP^{CM}). Computer-based training products, videos, and self-study workbooks are available through NAECs, Novell resellers, and Novell After Market Products.

For more information or to find the NAEC or NAEP nearest you, in the United States and Canada call 1-800-233-EDUC. In all other areas, call 1-801-861-5508, or contact your nearest Novell sales office.

A quick 24-hour-a-day tool for information is Novell Education FaxBack. Call 1-801-861-5363 for access to FaxBack. Ask for document 1448 for current courses. Ask for additional documents listing the NAEC in your area and the courses they offer.

BrainShare

Some of the best material presented about NetWare 4 and other Novell products are presented during Novell's BrainShareSM conference. BrainShare conferences are held in Salt Lake City, Utah; Europe; Japan; and Australia.

For more information in the United States and Canada call 1-800-NETWARE. In all other areas, call 1-801-861-5508, or contact your nearest Novell sales office.

Seminars

Helpful seminars are held in the local Novell sales offices. Your local account representative can assist you.

Novell Consulting Services Workshops

Novell Consulting Services (NCS) develops and delivers workshops to our Consulting Partners. If you are not a Consulting Partner, contact

Novell Consulting Services for an invitation to attend the workshops for a fee.

NCS provides a copy of the NCS Toolkit to all attendees of the workshops.

The Toolkit provides design and implementation methodologies for all Novell products. Tips and tricks that were learned through site visits and testing are also included.

AppNotes

Current AppNotes™ are a good source of NetWare 4 information. Older AppNotes that are based on NetWare 4.0, 4.01, 4.02, and 4.11 should be discarded.

For more information in the United States and Canada call 1-800-377-4136 or (303) 297-2725. In all other areas, call (303) 297-2725, or contact your nearest Novell sales office.

NUI Groups

Joining the local NetWare User Group allows you to share information and experiences with other NetWare 4.1x users. Novell engineers and consultants are often invited to speak on topics of interest.

For more information in the United States and Canada call 1-800-228-4684. In all other areas call (801) 228-4538, or contact your nearest Novell sales office or NetWare Users International group.

Where to Go from Here

To	Go to
Begin your NetWare 4 project	Chapter 2, "Gathering Information and Defining the Project Scope," on page 15

Gathering Information and Defining the Project Scope

Information about your organization's workflow, communication layout, organizational structure, and resources will help you identify the scope of the design project and framework of the design.

You will need to gather information about your organization. This information will be reference material for determining the best approach to rights structures, resource access, performance tuning, and management for your NetWare® 4™ implementation.

The following topics are discussed in the chapter:

- ◆ “Determining What Information Is Needed” on page 15
- ◆ “Defining the Project Design Scope” on page 20

Determining What Information Is Needed

Team members need to determine what information will best support the decisions they need to make. The information should be as current and detailed as possible. This ensures that your design will match your organization's real implementation of resources and workflow.

You should gather the following types of documents and information:

- ◆ Organization chart
- ◆ Resource lists
- ◆ Workflow information
- ◆ Location maps
- ◆ Maintenance schedules

- ◆ LAN/WAN topology maps
- ◆ Configuration information

MIS Manager

The following table list the type of information needed by the MIS manager:

Information Type	Purpose
Organization chart	Shows how the organization is structured.
Resource lists	Lists all the network resources that need to be incorporated into the Directory tree.
Workflow information	Helps determine how to best accommodate the organization's workflow so that little or no downtime is experienced.
Location maps	Shows how resources are allocated throughout the organization and where those resources are located.
Maintenance schedules	Helps determine when specific network tasks are being performed, such as backup and software and hardware upgrades.

NDS Expert

The following table lists the type of information needed by the NDS expert:

Information Type	Purpose
Organization chart	Identifies the major divisions and workgroups within the organization.
WAN topology	Determines how many sites exist in the organization and what types of WAN links exist between sites.
LAN topology	Identifies what major resources exist and where they are located. Helps determine wire type and configurations.
Location maps	Helps identify regional areas.
Workflow information	Helps determine how information flows through the organization.

NetWare Server Specialist

The following table lists the type of information needed by the NetWare server specialist:

Information Type	Purpose
WAN topology	Shows how many servers exist and where they are located. Helps identify what function NetWare servers perform for WAN communications.
LAN topology	Shows how many NetWare servers are in a given LAN segment. Helps identify servers with unique configurations.
Location maps	Identifies regional areas.
Resource lists	Lists all the NetWare servers and their locations. Helps determine what type of client support is needed and what versions of software exist or are needed.
Configuration information	Lists specific information about each NetWare server's specific hardware configuration, settings, and software version.

Workstation Specialist

The following table lists the type of information needed by the workstation specialist:

Information Type	Purpose
Resource list	Identifies how many workstations need to be upgraded and what type of hardware exists.
Configuration information	Lists specific information about each workstation's hardware configuration, settings, and software version.
WAN topology	Shows how many sites exist in the organization and what type of access is needed to the network.
LAN topology	Identifies how many workstations exist on a specific LAN segment and what wire type and configurations are used.

Information Type	Purpose
Location maps	Identifies regional areas.
Workflow information	Helps determine when and how workstations access and use information.

Application Specialist

The following table lists the type of information needed by the application specialist:

Information Type	Purpose
Resource list	Identifies how many licenses are needed to support each platform. Helps determine if software versions need to be upgraded.
LAN topology	Identifies which servers are used to store applications and how the servers are accessed.
Configuration information	Shows specific information about each application's configuration, settings, and software version.
Workflow information	Helps determine which applications are needed at which locations.

Printing Specialist

The following table lists the type of information needed by the printing specialist:

Information Type	Purpose
Resource list	Helps determine how many printers need upgraded drivers.
LAN topology	Identifies where printers are located and what workgroups access each printer.
Configuration information	Shows specific information about each printer's configuration, settings, and driver version.

Connectivity Specialist

The following table lists the type of information needed by the connectivity specialist:

Information Type	Purpose
Resource list	Identifies which protocols should be supported.
LAN topology	Helps identify how the LAN is routed to determine if a more efficient method can be used.
WAN topology	Helps determine the speed of WAN links between each site. Identifies how often remote sites dial in.
Location maps	Shows how information is being routed and if a more efficient method can be used.
Configuration information	Shows specific information about each application's configuration, settings, and software version.
Workflow information	Helps determine what applications are needed at which locations.

Testing Lab Coordinator

The following table lists the type of information needed by the testing lab coordinator:

Information Type	Purpose
Resource list	Identifies typical server and workstation configurations and setup.
LAN topology	Helps identify the kinds of routing LAN configuration that need to be simulated.
WAN topology	Helps identify the kinds of WAN links that need to be simulated.
Configuration information	Helps determine the types of configuration that need to be simulated.
Workflow information	Shows what applications and resources are being used.

Education and Training Coordinator

The following table lists the type of information needed by the education and training coordinator.

Information Type	Purpose
Organization chart	Shows which groups need training and identifies key contacts for scheduling training.
Resource lists	Shows what training needs to be done to use new software.
Workflow information	Helps determine current processes and how those processes can be improved by the new technology.

Defining the Project Design Scope

Before starting your NetWare 4 design, determine the project scope and timelines. The complexity of your organization's setup will determine the size and length of the NetWare 4 design phase.

To determine the scope of this project, ask yourself:

- ◆ How long will it take?
- ◆ Are we going to implement the design throughout the organization?

Identifying Critical Factors

Critical factors for evaluating the scope of the NetWare 4 design include:

- ◆ Determining complexity
- ◆ Identifying expectations

Determining Complexity

The following table reviews the factors that help determine the complexity of your organization:

Factor	Simple	Complex
Custom partitions	Fewer than 5 servers, single site	More than 5 servers, multiple sites
Time synchronization	Fewer than 29 servers, single site	30 or more servers, multiple sites
Number of locations	One	Multiple
Number of servers	Fewer than 5	Multiple
Number of users	Fewer than 1000 objects/users	More than 1000 objects/users
Setting up a testing lab	No lab	Set up lab
Organizational vs. departmental implementation	Add time to planning process if expecting growth or merger	Add time to planning process if expecting growth or merger

If you have a simple network, you should accept default settings in most cases. If you have a complex network, review the following list to determine the implications for each criteria:

- 5 or more servers
 - ◆ Design custom partitioning
 - ◆ Design custom replication
- 30 or more servers
 - ◆ Design time synchronization
 - ◆ Set up test lab before implementation
- Multiple sites
 - ◆ Design partitioning
 - ◆ Design time synchronization

- ❑ Multiple Directory trees in your organization
 - ◆ Determine who shares resources
 - ◆ Identify possibility of merging
- ❑ Multiple NetWare versions throughout the entire network
 - ◆ Involve several groups to plan and implement across the entire organization
 - ◆ Plan thorough integration testing of applications, utilities, hardware, etc.
- ❑ Possible growth of sites and number of network servers
 - ◆ Perform a more complex design
 - ◆ Plan for a more complex implementation

Identifying Expectations

Review the complexity and determine the expectations for the design project, including

- ◆ How long will the project last?
- ◆ What is the budget?
- ◆ Who will be on the project team?
- ◆ Will the implementation be across the whole organization or select areas?
- ◆ Are there any other critical needs from management?

Creating a Design Phase Schedule

Part of the project approach includes listing the tasks to be performed in the Design phase and estimating timelines for completing them. Use the procedures in the Design phase to create your timeline.

Creating New Documentation

Examples of planning documents are provided to help you manage and implement NetWare 4 on your network. You should customize these examples to fit your specific network environment.

See “Template Examples” on page 251 for more information.

Where to Go from Here

To	Go to
Organize network resources into logical units within a Directory tree	Chapter 3, “Designing the Directory Tree Structure,” on page 25
Scale the size of your network for better accessibility and fault tolerance	Chapter 4, “Determining a Partition and Replication Strategy,” on page 67
Establish a method for ensuring that network time is consistent among all network servers	Chapter 5, “Planning the Time Synchronization Strategy,” on page 89
Determine an efficient plan for accessing network resources	Chapter 6, “Creating an Accessibility Plan,” on page 107

Designing the Directory Tree Structure

This chapter describes the process for designing a Directory tree. The following topics are discussed:

- ◆ “Creating a Naming Standards Document” on page 30
- ◆ “Planning the Directory Tree Structure” on page 37
- ◆ “Planning Placement of Network Resources” on page 52

Introduction

The Novell® Directory Services™ (NDS™) technology allows you to develop a common, distributed Directory database of logical and physical resources regardless of where the resources are physically located—forming a single information system.

Hint: The term *Directory* used in Novell Directory Services is capitalized in this document to differentiate it from the term *directory* used in the file system.

Applications that have traditionally maintained Directory databases such as e-mail, human resources, and network management applications can now share a common source of Directory information that is available from any server in the network. This greatly enhances access to and management of resource information for the entire network.

All NetWare® 4™ servers on the network access the Directory database for information about network resources and access to those resources. Because of this, network information is not server specific. Each network resource is represented by a unique entry in the database and is accessed by its name, not by its association to a particular server.

All network clients access information and resources through a single connection to the network.

Note: Novell Directory Services (NDS) is consistent with the international standard, X.500. Some implementations of the X.500 standards within NetWare 4 may differ from those described by CCITT (Consultative Committee for Telegraphy and Telephony). This is because many of specifications were being defined after Novell completed development of NDS.

Object-Oriented Database

NDS is an object-oriented implementation of a Directory that allows you to build sophisticated naming schemes and logical representations of network resources. The objects used in NDS are referred to as *Directory objects*.

Directory objects are structures that store information; they are not the actual entity represented by the object. For example, a Printer object stores information about a specific printer and helps manage how the printer is used, but it is not the actual printer itself.

The specific information that is stored for each object then becomes the directory information that can be used by applications and resources across the entire network. The Directory information is stored in the *Directory database*.

Directory objects consist of categories of information, known as *properties* or *attributes*. These properties act as data fields within an object for defining specific information about the object.

Some properties contain vital network information, such as printer names, network addresses, and configuration information. Other properties contain non-vital information, such as a user's job title, telephone number, and street address. The specific property information is referred to as an object's *property value*.

Some property values are required. An object cannot be created without these values, and these values cannot be deleted. For example, a User object requires both the name of the object and a value for the user's last name.

Many properties are multivalued. This means that more than one value can be set for a given property of an object. For example, values for multiple phone numbers can be set in the Telephone Property of a User object.

The following figure illustrates the properties of a User object.

Object	Property	Value
User	Login name	Esayers
	Email address	Esayers@ACME
	Telephone number	555-1234 551-4321

Object Classes

The Directory database contains three types or classes of objects:

- ◆ [Root] object (Directory tree name)
- ◆ Container objects
- ◆ Leaf objects

These objects represent both actual and logical resources in the network, such as users, printers, groups, and print queues.

Directory Rules and Definitions

NDS maintains a system of rules and definitions for object properties that establishes the content and format of each object. This system of rules and definitions is called the *Directory schema*.

Base Schema

The basis for all entries in a Directory database is a set of defined object classes referred to as the *base schema*. Object classes such as servers, users, and print queues are some of the base object classes defined by the base schema.

Schema Extension

The NDS schema can be modified and extended to suit the specific needs of your organization. Object class definitions can be added to and modified for the existing base schema. This is consistent with the X.520 specifications defined by CCITT. You can extend the NDS base schema

by using Novell's Directory Application Programming Interface (API) to create new objects, class definitions, and properties as well as by adding properties to existing class definitions. For example, you may need a special object to represent a group of compact discs that could be reflected on all NDS servers in the tree.

Note: The complete list of flags associated with each property can be found in the *Novell Directory Services Schema Specification*, which is part of the NetWare server SDK.

However, the Novell 4.11 utilities will not allow you to add or modify objects in your new object classification. You will need to create utilities that can manage these new objects along with the currently defined NDS objects.

Directory Tree Structure

Directory objects are organized within the database by creating a hierarchical structure with multiple levels of organizational units, users, groups, and other network resources. This hierarchical structure is referred to as the *Directory tree*.

Note: The Directory tree replaces the bindery (flat file system) that was used in previous versions of NetWare.

The way in which objects are created and structured in the Directory tree is determined by the physical and organizational environments of your network.

Directory Tree Design

The size of your network isn't as important as the environment your network exists in—the network's hardware, communication links, LAN/WAN topology, and your organization's structure.

The complexity of your network's physical and organizational infrastructure determines the amount of designing necessary to efficiently implement the NDS technology—the more complex your environment, the more designing you need to do.

For example, a small network with multiple WAN links might have many more design concerns than a large network with no WAN links. This is because of unique physical attributes associated with different WAN architecture types.

Generally, four types of network environments affect the design of a Directory tree:

Network Environment	Description
Single server network	Typically maintains only LAN connections. The entire network is housed in one building or office and all connections lead to a single server.
Single location network	Typically maintains only LAN connections. The entire network is housed in one building or office.
Single campus network	Typically maintains only LAN or high speed WAN connections. The entire network is located at one site, in one or more buildings.
Multiple campus network	Typically maintains connections between campus sites. It relies on data transmission across LAN and WAN links, with issues concerning speed and volume.

The Directory tree design is the most important procedure in the design phase. All other NetWare 4 design is based on the tree structure that you create. Nevertheless, NDS is extremely flexible and will allow modifications to the Directory tree structure as your organization changes.

An efficiently designed tree

- ◆ Ensures that NDS object values are consistent, which makes networks easier to access and manage.
- ◆ Makes the successful design of partitions and replicas possible.
- ◆ Accommodates growth of the network without exacting revisions.
- ◆ Makes merging Directory trees easier.
- ◆ Makes the designing of other network services and accessibility easier.
- ◆ Makes navigating the tree more intuitive.
- ◆ Increases efficiency of network resources because they are available from anywhere in the network with a single login.

Objectives

Designing a Directory tree structure consists of creating a NDS naming standard and designing the Directory tree to best suit your organization's network environment. You will use the information that you gathered about your organization's workflow, communication layout, organizational structure, and resources to determine the actual framework of your tree design.

Prerequisites

- You should have copies of the following planning documents:
 - ◆ Organizational charts
 - ◆ Resource maps
 - ◆ Location maps
 - ◆ LAN and WAN topology maps
 - ◆ Workflow information
- You must have organized a project team
- Each project team member must be educated about NetWare 4 concepts

Creating a Naming Standards Document

Naming standards detail the conventions you will use for naming Directory objects. Standards should also specify how you will enter property values (such as telephone numbers and addresses) for the objects.

You should create a document and distribute it to those responsible for adding or moving objects in the Directory tree. If standards are in place for using the Directory, users and administrators can better use the Directory tree.

Searching and browsing rely heavily on the ability to search the Directory based on criteria from the user. If the object names follow a standard, then searching is simpler.

For example, if all laser printers are named *LJunique name*, where *unique name* is more descriptive, then a search for all printers named *LJ** is feasible.

Using a Naming Standards Template

Use the naming standards template example to determine an appropriate naming standards document for your organization

The following example provides standards and rationale for developing a naming standards document.

Table 3-1
Naming Standard Example

Item	Standard	Examples	Rationale
User object common name (login name)	First initial, middle initial (if applicable), last name (all lower case), 8 characters maximum. All common names are unique in the company	msmith, bgashler	Using unique names company-wide is not required by NDS but helps avoid conflicting names in the same context. An 8-character maximum simplifies user home directory creation.
User object last name	Last name (lowercase)	poulton	Using lowercase for the last name helps distinguish it as the last name of a user.
User object telephone and fax	Numbers separated by dashes	US: 1234567890 Other: 44344123456	Avoid parentheses, commas, or the long-distance number 1-. This field is expected to be used by autodialing software.
User object location	Two-letter location code (uppercase), dash, mail stop	BA-C23	This field is used by interdepartment mail carriers.
Directory tree	For the main corporate tree. Use a location name for other trees if necessary.	ACMECORP	Separate trees may be created for sites where connection to the corporate tree is not economical.
Organization	ACME for all trees	ACME	A standard Organization name allows for future merging of trees.

Item	Standard	Examples	Rationale
Organizational Units whose names are based on a major location	Two-letter location code	AT, CH, CU, LA, BA, BO, DA, MU	Short, standard names are used for efficient searching.
Other OU names	Department name or abbreviation	Sales, Eng	Short, standard names are used for efficient searching.
Common names of leaf objects other than users	Unique company-wide One-letter object class, dash, two-letter location code, brand or department information, unique number	LaserJet 4 printer in Chicago: P-CH-LJ4-023 Engineering server in Chicago: S-CH-Eng-1	Avoid extremely long names; some utilities will not display them. NDS requires server names to be unique in the tree.
Special objects			Short, standard names are used for efficient searching.
◆ Organizational Role	Administrative role the Organizational Role performs	PrintAdmin	
◆ Profile	Name should reflect the purpose of the profile	MobileUser	
◆ Directory Map	Directory name where application has been installed	DOSAPPS	
All	Avoid special characters such as : . + = / \ Avoid spaces		Special characters are not allowed in utilities. Spaces demand the use of quotation marks in some utilities.

See Appendix C, "Template Examples," on page 251 for a copy of a naming standards document template.

Identifying Naming Considerations

Concise and meaningful object names make it easier to use the Directory tree. Keeping names short reduces the amount of data going across the wire, simplifies logins, and makes names easier to remember. To ensure that object names are concise, consider the following:

- ◆ Consistency
- ◆ Name length
- ◆ Compatibility
- ◆ Conventions

Consistency

Consistent naming standards provide a guideline for network supervisors who will be adding servers, creating users, modifying and moving objects, etc. Consistent standards also make it easy for users to search for specific items in the Directory tree.

Hint: Although a consistent naming standard for the corporate network is important, you do not need to have it perfected before you implement NDS. You can rename objects and move subtrees to reflect any changes you want to implement.

Name Length

Ensure that the naming schemes are short, yet as descriptive as possible. For example, SW Engineering could be shortened to SWEng.

All object names can contain up to 64 characters in their Name property (the name given when an object is created).

Compatibility

Some compatibility issues exist that you should consider, such as

- ◆ Backwards compatibility

When you create objects to be accessed from a client workstation running the Novell Client™ software, the names of the objects

must follow bindery naming rules or the Novell Client software cannot recognize them. Object names in bindery services are interpreted as follows:

- ◆ Spaces in object names are replaced by underscores
- ◆ Object names are cut off after the 47th character

You cannot use the following characters in an object name that must be accessed from a client running a version of NetWare earlier than NetWare 4:

Character	Description	Character	Description
/	Slash	[]	Square brackets
\	Backslash	< >	Angle brackets
:	Colon		Vertical bar
;	Semicolon	+	Plus sign
,	Comma	=	Equal sign
*	Asterisk	?	Question mark

Note: The object naming rules apply to most objects. See the documentation provided with your application to obtain specific naming rules.

- ◆ DOS commands

Avoid using spaces because they make it more difficult for users to reference objects from the DOS environment. If you want to use a space in names, use an underscore character (_) instead of a space. NDS treats spaces and the underscore character interchangeably.

- ◆ International

Unicode* is a wide character encoding scheme that provides the basis for internationalization of the information in an NDS database. All character strings exchanged between a NetWare 4 server and a client workstation are in Unicode. The Novell Client Software handles the translation of Unicode strings.

Occasionally, however, you might use characters that Unicode cannot translate. When this happens, the character is substituted in your display as a heart symbol in DOS and as a box symbol in Windows*.

Substituted characters can prevent NDS from recognizing an object. See “Code Page” and “Unicode” in *Concepts* for more information.

◆ IPX numbers

You may want to set standards for IPX internal and external network numbers. This makes setting up servers more difficult, but it can simplify troubleshooting and packet filtering.

Type	Explanation
IPX external network number	<p>The IPX external network number consists of eight hexadecimal digits (0 through F) which identify the cable segment. A new IPX external network number is assigned to each segment for additional protocols or frame types.</p> <p>The IPX external network number is specified in a server's AUTOEXEC.NCF file by the BIND parameter NET=. When you use INETCFG.NLM to configure protocols, the number is called the ID String.</p> <p>You can set standards for the IPX external network number by assigning codes for the digits. For example, you could specify that all cable segments in Chicago begin with the digits 01. The remaining digits could further narrow the location of the cable segment or the type of protocol.</p>
IPX internal network number	<p>The IPX internal network number is also an eight-digit hexadecimal number. It is specified in the server's AUTOEXEC.NCF file to uniquely identify the server on the network.</p> <p>The server installation program normally generates a unique IPX internal network number, though it does allow the installer to specify one.</p> <p>Like the external number, you can set standards for the internal number to help locate the source of packets during troubleshooting. For instance, you could specify that all cable segments in Chicago begin with the digits 01. The remaining digits could further narrow the location or type of the server.</p>

- ◆ NetWare Server objects

The NetWare Server object for a NetWare 4 server must be created with INSTALL. The object is given the same name as the physical server. To see rules for naming physical servers, press <F1> (Help) in INSTALL.

If you create a NetWare Server object for servers that are not running NetWare 4, you must ensure that the object name is the same as the server name. This is because NDS searches for the server by name to verify its existence.

Hint: Because of these restrictions, we recommend renaming NetWare Server objects by changing their names in the AUTOEXEC.NCF file.

- ◆ Service Advertising Protocols (SAP)

Unique names are required for all devices on the network which send out Service Advertising Protocols (SAP). File and print server names, even though they are stored in NDS, need to be unique because they use SAP.

Conventions

Use the following guidelines when creating your naming conventions:

- ◆ Use consistent capitalization to ensure readability and to differentiate between container objects and leaf objects.
- ◆ To facilitate administration when working at the command line, avoid using spaces. Use hyphens or underscores instead.
- ◆ To ensure compatibility with bindery services, limit names to 47 characters, and do not use the following characters in object names:

Character	Description	Character	Description
/	Slash	[]	Square brackets
\	Backslash	< >	Angle brackets
:	Colon		Vertical bar
;	Semicolon	+	Plus sign

Character	Description	Character	Description
,	Comma	=	Equal sign
*	Asterisk	?	Question mark

- ◆ Assign user login names using the first initial of the first name and then up to seven characters of the last name. For example, user John Smith would have a login name of JSmith.

Planning the Directory Tree Structure

Novell Directory Services supports many different Directory tree structures. This flexibility ensures that your tree design is the most efficient for your particular network environment. To ensure that you build the most efficient structure, you should maintain four goals:

- ◆ Provide intuitive and efficient access to network resources.
- ◆ Provide secure access to shared resources that is easily maintained and monitored.
- ◆ Develop a clear and concise blueprint for completing the implementation process.
- ◆ Develop a flexible layout of the tree structure to ensure that future changes can be easily incorporated.

The basic design principles provided in this section should assist you in accomplishing these goals.

You should build your tree structure by actually drawing each layer of the Directory tree as you read the following discussions. You can use modeling or drawing software, or create diagrams of the structure on paper.

The most helpful documents for this procedure are organization and resource charts, location maps, and WAN topology charts.

Designing Upper Layers

Structuring the upper layers of the Directory tree is important to the general foundation of the tree and the tree performance. The upper layers are mostly affected by the WAN topology and speed of WAN links. This affects the kind of partitions and replicas you can use.

The upper layers include the following objects:

- ◆ [Root] object
- ◆ Organization objects
- ◆ First layer of Organizational Unit objects

These layers build a solid framework for the bottom layers to be built upon. A few users and network resources need to be placed in the upper layers of the tree.

Naming Your Directory Tree

The highest object in the Directory tree is the [Root] object and is given the tree name. The [Root] object resides at the top of the tree and all other objects branch downward from it.

The [Root] object can be created only by the NetWare 4 installation program, which automatically places it at the top of the tree. Once the [Root] object is named, it can only be renamed by the DSMERGE utility.

A Directory tree name must be unique. Use a name that identifies your network and the organizational environment that the tree is designed for.

For example, a company that is named ACME Corporation and maintains a world-wide network might simply name their Directory tree ACME_CORP.

Important: Multiple Directory trees can exist on the same network, but each tree needs a unique name.

Workstations use the tree name to connect to a server within the correct Directory tree. Network utilities reference the tree name by its unique name or by the [Root] object. Many utilities only refer to the object as [Root].

The Directory tree name or [Root] object can have trustees, and the rights granted to these trustees flow down the tree. One example is the User object ADMIN, which is created automatically during installation.

Determining the Directory Tree Foundation

The general foundation of the Directory tree is established through container objects. These objects represent the organizational and physical structure of the network.

Container objects hold (or contain) other Directory objects. Container objects provide a means of logically organizing all other objects in the Directory tree.

There are five kinds of containers:

- ◆ Country (C)

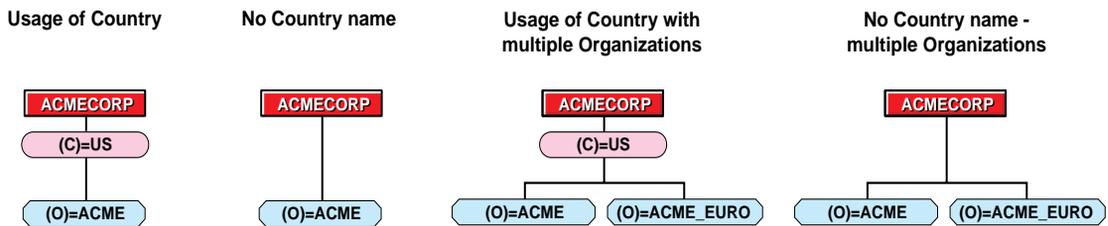
The Country object designates the country where your network resides and organizes other objects within the country. This object is consistent with the X.500 specifications and is generally used by global directory providers as an entry point into their directory systems.

Hint: Novell Directory Services more commonly uses Organizational Unit objects to represent geographical boundaries within an organization.

If you are not planning to use a third-party Directory provider, using the Country object might add an unnecessary level of complexity. If in the future the Directory tree needs the Country designator or needs the object added for merging with another tree, you can then add the Country object.

When used, the Country object must be placed directly below the [Root] object.

Figure 3-1
Example of a Tree Design with and without a Country Object



- ◆ Locality (L)

The Locality object designates the location where this portion of your network resides and organizes other objects within the location.

The Country and Locality objects are characteristic of the X.500 specification, but are not commonly used within the Novell Directory Services design. This chapter does not discuss design information for using Country or Locality objects.

Warning: Many NetWare 4 utilities do not recognize the Locality object. Use this object only when necessary for compliance with the X.500 specifications.

When used, the Locality object can be placed below the [Root] object, Country object, Organization object, or Organizational Unit object.

- ◆ Licensed Product (LP)

Licensed Product objects are created automatically when you install a license certificate or create a metering certificate using NetWare Licensing Services™ technology.

When an NLS-enabled application is installed, it should add a Licensed Product container object to the Novell Directory database and a License Certificate leaf object to that container.

When used, the Licensing Product object can be placed below the [Root] object, Country object, Organization object, or Organizational Unit object.

For more information, see “Managing NetWare Licensing Services” in *Supervising the Network*.

- ◆ Organization (O)

- ◆ An Organization object helps you organize other objects in the Directory tree. It also allows you to set defaults for User objects you create in the Organization container.

You can use an Organization object to designate a company, a division of a company, a university or college with various departments, or a department with several project teams.

Every Directory tree must contain at least one Organization object.

Organization objects must be placed directly below the [Root] object, unless a Country object or Locality object is used.

- ◆ Organizational Unit (OU)
- ◆ An Organizational Unit object helps you organize lower layer containers and other objects in the Directory tree. It also allows you to establish system level login scripts and to create a user template for User objects you create in the Organizational Unit container.

You can use an Organizational Unit object to designate a business unit within a company, a department within a division or university, or a project team within a department.

This object is optional. When used, Organizational Unit objects must be placed directly below an Organization, Organizational Unit, or a Locality object.

Establishing Tree Boundaries

Create container objects to establish a logical representation of the organizational and physical network infrastructure. This allows you to establish tree boundaries that facilitate efficient administration, scalability, security, and access to network resources.

The most helpful documents for this task are organization charts, location maps, and WAN topology charts.

Forming an Organization Layer

Novell Directory Services requires that at least one Organization object exist in the Directory tree. Organization objects can reside directly under the [Root] object or under a Country or Locality object.

Generally, a single Organization object is created directly under the [Root] object. It identifies an organization's name and provides a level of administration for the entire tree.

For example, a network administrator might set a specific set of file system rights for all files on network servers, or establish a system login script that is used for all users within an organization or department.

A single Organization object under the [Root] object also allows for easier merging of trees with other organizations.

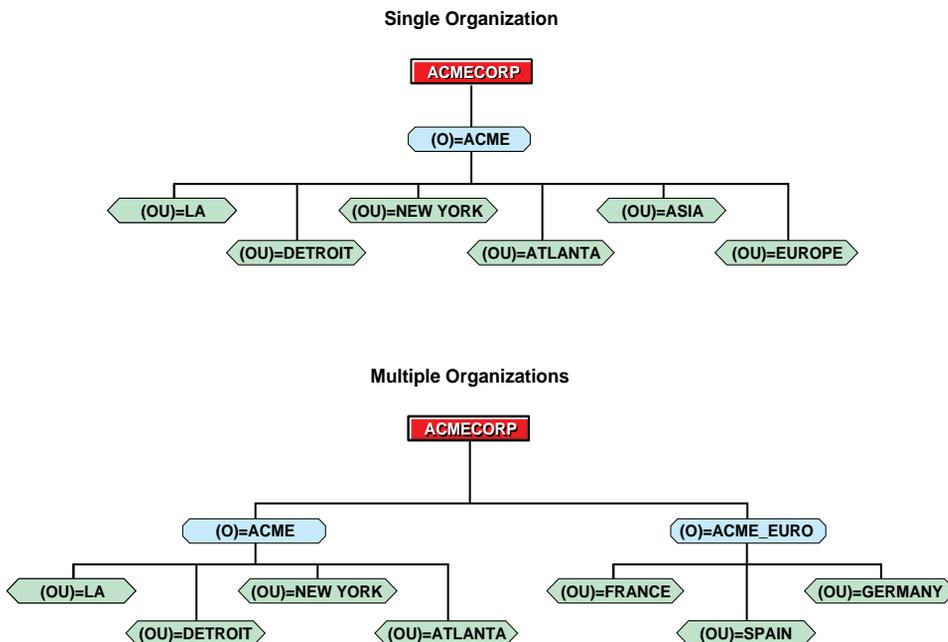
If the Directory tree supports multiple organizations that maintain distinct administration strategies or act as separate business units, you may want to create multiple Organization objects.

For example, if the ACME Corporation functions as a single business unit and maintains the same general strategies for administration, access, and file system rights for the entire organization, create a single Organization object and name it ACME.

If the ACME Corporation functions as two distinct business units without common rights or access between the general business headquarters and the manufacturing headquarters in Europe, create two Organization objects named ACME and ACME_EURO.

The following figure illustrates how to use two Organizational object containers.

Figure 3-2
Directory Tree Design with Single and Multiple Organizational Object Containers



Use a name representative of the organization for naming the Organization object. Once you establish an Organization object name, consider registering the name with either the Internet or ANSI to enable future X.500 multiple tree and multiple name space operations.

When naming an Organization object, use short, concise names. Short names provide easy access to network resources and simple navigation throughout the tree. Creating short names also reduces the likelihood of mistyping names.

Forming Regional and Location Layers

The upper layers of the Directory tree are most affected by the geographical and physical structure of your network environment.

Because of this, you should structure the upper layers of your Directory tree to represent the major geographical locations in your organization and the network's hardware, communication links, and LAN/WAN topology. Doing this allows you to scale the Directory database according to the network's WAN structures.

The following table describes how to use Organizational Unit objects to represent a region-based and/or location-based structure of your organization's geographical network.

Type	Purpose
Location-based	<p>Use the geographical locations within your organization's physical network infrastructure for naming location-based Organizational Unit objects.</p> <p>For example, if you have departments in different parts of a region, city, or campus, you could create SJC, SND, SFO containers, NORTH, SOUTH, EAST, WEST containers, or BUILD1, BUILD2, LIBRARY, HQ.</p> <p>Place the locational Organizational Unit objects directly under the Organization object layer.</p> <p>The locations should be geographical sites that have multiple departments or divisions included at each site. If the locations are small remote offices or field offices, you should place locational objects for those offices under an Organizational Unit object at a lower layer in the tree. See "Designing Lower Layers" for more information.</p>

Type	Purpose
Region-based	<p data-bbox="600 181 1212 352">Complex networks with multiple campus sites contained within a specific region should maintain a layer of region-based Organizational Unit objects above the location-based Organizational Unit objects. The network is easier to administer if the organization has regional offices that each manage a number of locations.</p> <p data-bbox="600 377 1204 518">For example, if you have divisions in three parts of a country and multiple departments in each part, you could create a WEST, MID, and EAST containers to contain individual groups of location-based containers, such as SJC, SND, SFO.</p> <p data-bbox="600 543 1204 684">Designing your tree according to the geographical structures of your organization also provides a more intuitive interface to the Directory tree. Users at a specific location can identify their location (or context) in the Directory tree by the location, or city that they work in.</p> <p data-bbox="600 709 1182 822">Organizing upper layers of the tree by the geographical structures in your organization also makes it easier for users to remember where their commonly-accessed network resources are.</p> <p data-bbox="600 846 1204 904">It also provides better organization for storing information and applications in your environment.</p>

Note: If you are designing a tree for a single site or a single or multiple campus environment with fast WAN links (T1 or better) and have no future plans for expanding to other locations, you won't have issues concerning the geographical or physical network structure. Go to "Designing Lower Layers" on page 46 to continue.

The following figures illustrate how to design a Directory tree according to your organization's geographical structure.

Figure 3-3
Physical Map of an Organizational Structure

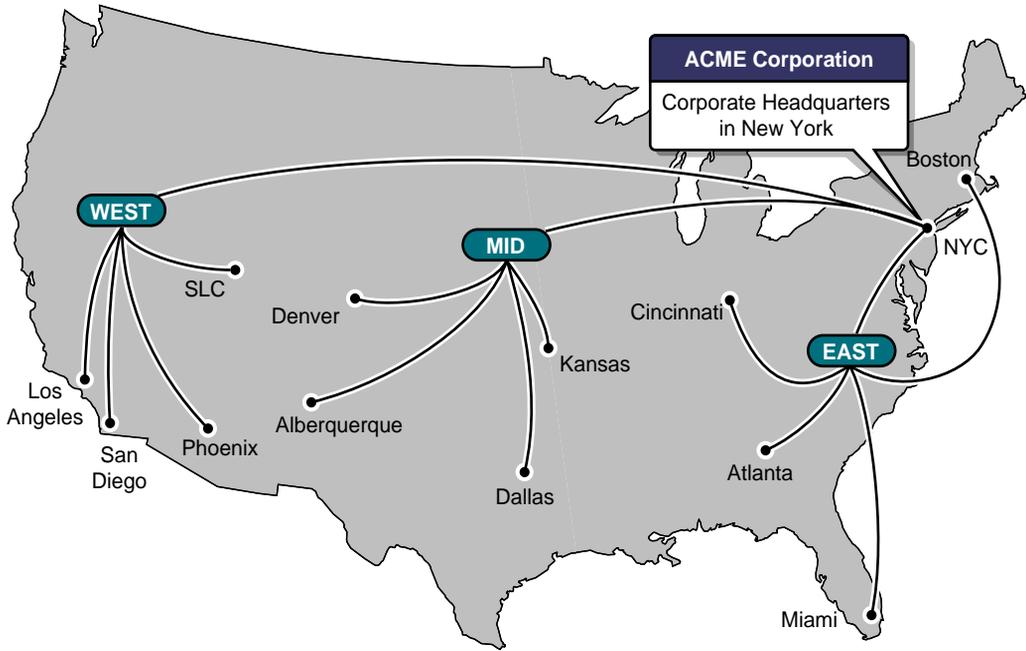
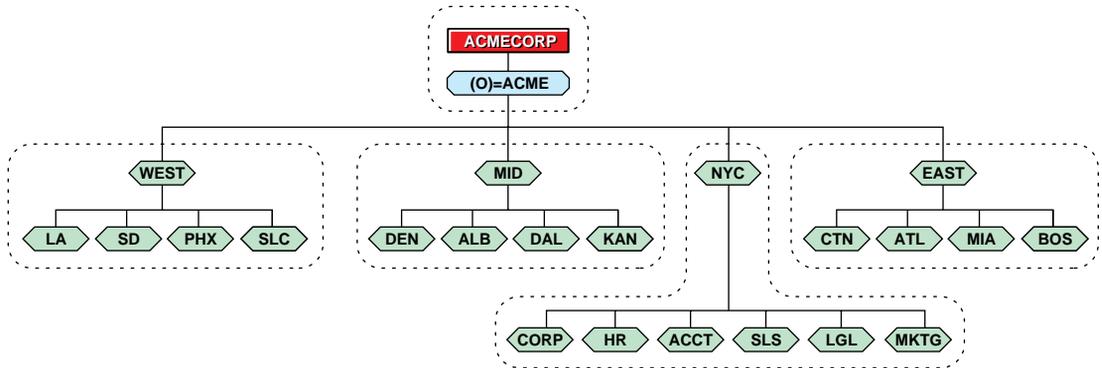


Figure 3-4
Example of Region-based Tree Design



Novell Directory Services is designed to operate in a fairly stable networking environment. The Directory tree design should account for any intermittent links, on-demand links, or WAN links with minimal available bandwidth that exist in the network.

When designing for particular LAN/WAN topologies, consider the following factors:

- ◆ Bandwidth
- ◆ Speed
- ◆ Cost
- ◆ Regulations

Designing Lower Layers

The lower tree layers provide a high level of flexibility to support any layout that suits your environment needs.

The lower layers include the following objects:

- ◆ Second and subsequent layers of Organizational Unit objects
- ◆ Leaf objects

These layers are built upon the foundation established in the upper layers. They provide the logical divisions of network resources and lower level organizational structures that exist in your organization. Most objects in your tree will be in these layers.

The most helpful documents for this task are organization and resource charts.

Determining the Directory Tree Framework

The general framework of the Directory tree is established through the Organizational Unit objects that represent the logical divisions, departments, and workgroups in your organization.

Organizational Unit objects help you to organize lower layer containers and other objects in the Directory tree. They also allow you to establish system-level login scripts and to create a template for User objects.

Lower layers should be designed for naming and organizing network resources and not for creating consistent subtrees or container structures within your tree.

The following two figures illustrate how container objects can be used to design the lower layers of your Directory tree structure.

Figure 3-5
Lower Layer of a Simple Directory Tree

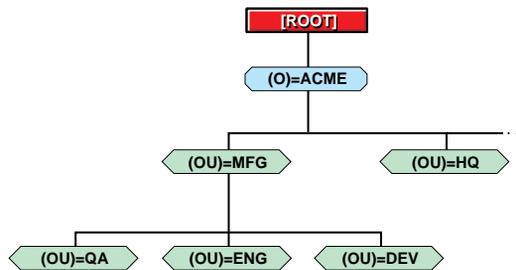
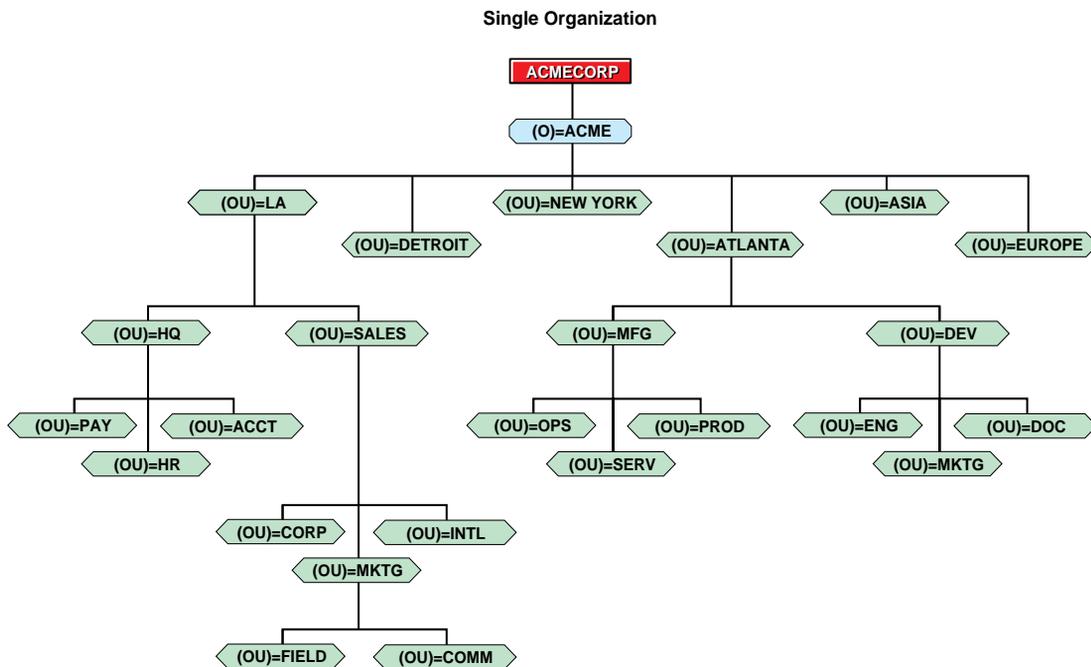


Figure 3-6
Lower Layer of a Complex Directory Tree



Forming Containers Based on Common Access

The most important consideration when structuring the lower layers is to create containers for network resources that have common access needs to other Directory objects. This allows you to make single trustee assignments or administer a single system login script for many users.

Use an organizational chart to provide the logical structure for container names.

Individuals in your organization chart should not be included in the tree as Organizational Unit objects. Container objects should also not be created as placeholders. Each container should only be created for existing functional groups and resources within the organization.

Use Organizational Unit objects to represent functional groups as container objects. Place these containers directly under the location-based Organizational Unit objects or under the Organization object

depending on how many geographical locations exist in your organization.

See Chapter 6, “Creating an Accessibility Plan,” on page 107 for more information.

Forming Containers Based on Workflow

Identify the workflow of your organization to determine possible groupings of individuals and services based on tasks rather than departmental lines.

If your organization maintains long-term projects or is highly process-oriented, you might create container objects.

Use Organizational Unit objects to represent the projects or products as container objects. Place these containers under Organizational Unit objects in a division or department. These types of containers are usually created at the lowest layers within the tree.

Use a project resource chart relying on project names to provide the logical structure for container names.

Individuals in your project chart should not be included in the tree as Organizational Unit objects. Container objects should also not be created for short-term projects or as place holders for future projects. Each container should only be created for project teams or production groups that access and use similar resources in the organization.

Forming Containers Based on Management Structures

The administrative model used in your organization can affect the design of the lower tree layers. With NetWare 4™ software, you can centralize network administration so that a single person or small group of people control the entire network.

You can also distribute administration so that many network supervisors throughout the organization control their own portion of the Directory tree.

The following table describes the three types of administrative models that should be considered when designing the lower layers of the tree.

Administrative Model	Action
Centralized	To provide for greater central administration of the network, design the tree with more layers of hierarchy for faster browsing.
Decentralized	To provide for better local administration of the network, design the tree with separate container objects for each administrative area in the tree.
Mixed	To provide for mixed administration of the network, use a hybrid of both central and local administration. You should ensure, however, that lower layer containers are managed by local administrators and upper layers are administered at central sites.

Forming Containers Based on Service Groups

Use a service-oriented approach such that users of similar services are grouped together across departments or other boundaries.

Incorporating Specific Design Elements

Some specific design elements may exist for your network that affect the framework used in the lower layers of the Directory tree. These elements are:

- ◆ Database scalability (partitioning and replication)
- ◆ Number of objects
- ◆ Network infrastructure
- ◆ Tree depth and name length

Database Partitioning and Replication

To adequately scale the Directory database and distribute portions of it across multiple servers, you should ensure that at least one Organizational Unit object exists for each portion of the tree that you want to partition.

Number of Objects

Create separate containers if the number of objects for a given container exceeds 5,000 objects. This provides for easier administration and scalability.

Network Infrastructure

Create a container for remote sites with slow links that belong to a particular division or department. For example, objects that represent resources belonging to a field sales office should be separated into their own container.

Tree Depth and Name Length

An appropriate depth for your environment enables easier access and management.

The Directory tree should maintain a depth between four to eight layers. As the complexity of your environment increases, either in numbers of objects or in the number of locations under single management, the tree depth will usually increase to accommodate those conditions.

You should also ensure that the accumulated name length from the lowest layer to the top of the tree ([Root] object) does not exceed 256 characters.

Limitation of DOS command line utilities impose a maximum name length of 255 characters. When shorter Organizational Unit names are used, a deeper tree may be designed. However, the greater the depth of the tree, the more complex access to network resources may be.

When objects are created, they are checked to ensure that the maximum name length of the object isn't exceeded. However, it is possible to later rename an object and cause the name to exceed 255 characters.

Reviewing the Directory Tree Structure

As you draw (model) the Directory tree structure, also evaluate the tree design. Each member of the project team should have the opportunity to review the tree structure before continuing to the next procedure for placing network resources in containers.

If you are upgrading from NetWare 3™, you can use the Novell Upgrade wizard to assist you in modeling your bindery data for migrating to the NetWare 4 Directory database.

Planning Placement of Network Resources

Intuitive, quick, and secure access is the most important factor in determining where network resources should be placed in the Directory tree. To ensure that you identify the most efficient placement of network resources, you should

- ◆ Establish consistent patterns for ensuring that objects reside near the resources that access them.
- ◆ Create processes to ensure that network administrators provide the same accessibility and security for all objects in the tree.
- ◆ Develop a clear and concise blueprint for completing the implementation process.

Place network resources in your tree structure by actually drawing each object in the container that it will reside in.

The most helpful documents for this procedure are the Directory tree structure drawing, resource charts, and administration maps.

Identifying Leaf Object Types

The network resources that are used in your organization are represented by objects called *leaf objects*.

Leaf objects are objects that do not contain any other objects. These commonly represent actual network entities such as users, servers, printers, and computers. You create leaf objects within a container object.

Note: If you are upgrading from a previous version of NetWare or migrating from a different operating system, the installation or migration utilities will identify many of the objects for you and place them in the container representing the server they were located on.

The following table lists the current set of leaf object types available in NetWare 4. You should review the list to identify which object types you can create from the base set of objects.

Table 3-2
Available Leaf Objects

Leaf Object	Description	When to Use
Alias	<p>Points to another object in the Directory tree and makes it appear as if the object that it points to actually exists in the Directory tree where the Alias object is.</p> <p>Although an object appears both where it was actually created and where an Alias referring to it was created, only one copy of the object really exists.</p> <p>If you delete or rename an Alias, the Alias itself (not the object it is pointing to) is deleted or renamed.</p>	<p>Use this object to allow access to an object that is in another context.</p> <p>For example, you can use an Alias to represent a resource, such as a special printer, that most users in the tree need to access.</p> <p>Also, when you move or rename a container object in a Directory tree, you have the option of creating an Alias object in place of the moved or renamed object. If you select this option, NetWare Administrator automatically creates the Alias for you and assigns it the same name as the original object.</p> <p>Creating an Alias object in place of a moved or renamed container object allows users to continue logging into the network and to see the container object (and the objects it contains) in its original Directory location.</p>

Leaf Object	Description	When to Use
Application	<p>NetWare Application Manager allows network supervisors to manage network applications as Application objects in the Novell Directory tree.</p> <p>NetWare Application Manager displays icons for all available applications in a window, and lets the user select an icon to launch an application. Users don't need to worry about drive mappings, paths, or rights.</p> <p>The network supervisor can manage applications at the container, Group, or User object level.</p>	<p>Application objects allow you to manage the network more efficiently, saving time and effort administering applications.</p> <p>Application objects simplify administrative tasks such as assigning rights, customizing login scripts, and supporting applications.</p>
Auditing File	<p>The Auditing File Object (AFO) is the Novell Directory Services data structure used to manage an audit trail's configuration and access rights.</p>	<p>An audit utility (such as AUDITCON) creates the AFO when you enable auditing. The server then checks for access rights each time a user attempts to access the audit trail.</p>
Computer	<p>Represents a nonserver computer on the network, such as a workstation or a router.</p>	<p>Use this object to store information about a nonserver computer, such as its network address, serial number, or person the computer is assigned to.</p> <p>This object has no effect on the operation of the network; it only stores information about the computer.</p>

Leaf Object	Description	When to Use
Directory Map	<p>Represents a particular directory in the file system. Directory Map objects can be especially useful in login scripts by pointing to directories that contain applications or other frequently used files.</p> <p>For example, if you have a directory that contains DOS 6.0, you will probably map a search drive to that directory in any login scripts you create. Later, if you upgrade to DOS 6.2 and rename the directory, you have to change the mapping in every login script where that search mapping appears.</p> <p>If you use a Directory Map object, you only change the information in that object, not in all login scripts.</p>	<p>Use this object to avoid making changes to many login scripts when the location of applications changes; instead, you change only the Directory Map object.</p>
Distribution List	<p>Represents a list of mail recipients.</p>	<p>Use this object to simplify sending mail to recipients.</p> <p>For example, you could create a Distribution List object called Recreation Committee. Anyone wanting to send a message to all the members in the Recreation Committee can simply address the message to Recreation Committee, rather than to each member.</p>

Leaf Object	Description	When to Use
External Entity	<p>Represents a non-native NDS object that is imported into NDS or registered in NDS.</p> <p>NetWare MHS Services use External Entity objects to represent users from non-NDS directories to provide an integrated address book for sending mail.</p> <p>MHS Services software is available on NetWare.</p>	<p>If your messaging environment contains non-MHS servers, such as SMTP hosts, SNADS nodes, or X.400 MTAs, you might choose to add users and lists at these servers to your NetWare database as External Entities.</p> <p>Adding these objects to the database as External Entities adds them to the address books of your messaging applications. When addressing messages, local users can choose non-MHS users and lists from a directory list.</p>
Group	<p>Assigns a name to a list of User objects that can be located anywhere in the Directory tree.</p>	<p>Create a Group when you have many User objects that need the same trustee assignments. Rather than making many trustee assignments, you make just one trustee assignment to all the users who belong to the group by making the trustee assignment to the Group object itself.</p>
LSP Server	<p>When you register an LSP (License Service Provider) with NDS, an LSP Server object is created, by default, in the same context as the NetWare Server object on which it is loaded. The LSP Server object can be relocated to another context in the Directory.</p> <p>An LSP Server object appears only if you load the NLS (NetWare Licensing Services) NLM program with an -r option.</p>	<p>A client-specific component packages the request and submits it to the nearest connected LSP.</p> <p>If the client is not connected to an LSP, the client checks the Novell Directory database, searching up the Directory tree for an LSP Server object.</p>

Leaf Object	Description	When to Use
Message Routing Group	Represents a group of messaging servers that can transfer messages directly with each other.	Create a Message Routing Group when you have two or more messaging servers that need to communicate with each other.
Messaging Server	Represents a messaging server that resides on a NetWare server. A Messaging Server object is automatically created in the Directory tree when you install NetWare MHS Services on a NetWare server. MHS Services software is available on NetWare.	Create a Messaging Server (by installing NetWare MHS Services on a NetWare server) when you need a server to handle messaging between users and groups on the network.
NetWare Server	Represents a server running NetWare. Store information about the server in the NetWare Server object's properties, such as the server's address, the physical location of the server, and what services it provides. In addition to storing information about the NetWare server, the NetWare Server object affects the network in that it is referred to by several other objects.	Use the NetWare Server object to tie the physical server to the Directory tree. Without this object, you cannot access file systems that are on that server's volumes. If you have a non-NetWare 4 server, you must create this object to be able to access those non-NetWare 4 volumes. When you create a NetWare Server object for a non-NetWare 4 server, the non-NetWare 4 server must be running.

Leaf Object	Description	When to Use
Organizational Role	Defines a position or role within an organization.	<p>Create an Organizational Role object so that you can assign rights to a particular position rather than to the person who may occupy that position. The occupant may change frequently, but the responsibilities of that position do not.</p> <p>You can assign any user to be an occupant of the Organizational Role object because every occupant receives the same rights that you granted to the Organizational Role object.</p>
Print Queue	Represents a print queue on the network.	<p>You must create a Print Queue object for every print queue on the network.</p> <p>This object cannot be created with NETADMIN. Use the NetWare Administrator or PCONSOLE.</p>
Print Server	Represents a network print server.	<p>You must create a Print Server object for every print server on the network.</p> <p>This object cannot be created with NETADMIN. Use NetWare Administrator or PCONSOLE.</p>
Printer	Represents a physical printing device on the network.	<p>You must create a Printer object for every printer on the network.</p> <p>This object cannot be created with NETADMIN. Use NetWare Administrator or PCONSOLE.</p>

Leaf Object	Description	When to Use
Profile	<p>Contains a profile login script. When the Profile object is listed as a User object's property, the Profile object's login script is executed when that User object logs in. The Profile login script executes after the profile login script and before the user login script.</p>	<p>Create a Profile object for a set of users who need to share common login script commands but who are not located in the same container in the Directory tree, or who are a subset of users in the same container.</p>
User	<p>Represents a person who uses your network.</p> <p>In the User object properties, you can set login restrictions, intruder detection limits, password and password restrictions, security equivalences, etc.</p>	<p>You must create a User object for every user who needs to log in to the network. When you create a User object, you can create a home directory for that user. When you create User objects, you can also choose to apply a template to the User object that provides default property values.</p> <p>For users who have NetWare 4 workstations, you can create the User objects anywhere in the Directory tree, but the users must know their context in order to log in. You should create User objects in the container where the users will typically log in.</p> <p>For users who have non-NetWare 4 workstations, you must create the User objects in the container where the bindery services context is set for the server that they need to log in to. (Bindery services is set by default for every NetWare 4 server that is installed.) Non-NetWare 4 users do not need to know their context because they log in to the server rather than to the Directory tree.</p>

Leaf Object	Description	When to Use
Volume	<p>Represents a physical volume on the network.</p> <p>In the Volume object's properties, you can enter identification information, such as the Host server, volume location, etc. You can also set restrictions for use of the volume, such as space limits for users.</p>	<p>You can create a Volume object for every physical volume on the network. (During installation of NetWare 4 on a server, Volume objects are created for every physical volume on that server.)</p> <p>Use INSTALL to create volumes on NetWare 4 servers.</p> <p>When a Volume object is created, the server name and the volume name information is placed in the Volume object's properties.</p> <p>You can use the Volume object to display information about the directories and files on that volume.</p>

Note: The Message Routing Group, External Entity, and Distribution List objects are NetWare Message Handling Service (NetWare MHS) objects. They appear in NetWare Administrator only if you have installed NetWare MHS Services on your NetWare servers.

Determining Leaf Object Placement

Determining where to place leaf objects in the tree depends on the container structure that was established. It may be necessary to change the container structure to accommodate some objects within your tree.

The following figures illustrate how leaf objects can be placed in the Directory tree structure.

Figure 3-7
 Example of Leaf Objects in a Simple Tree Design

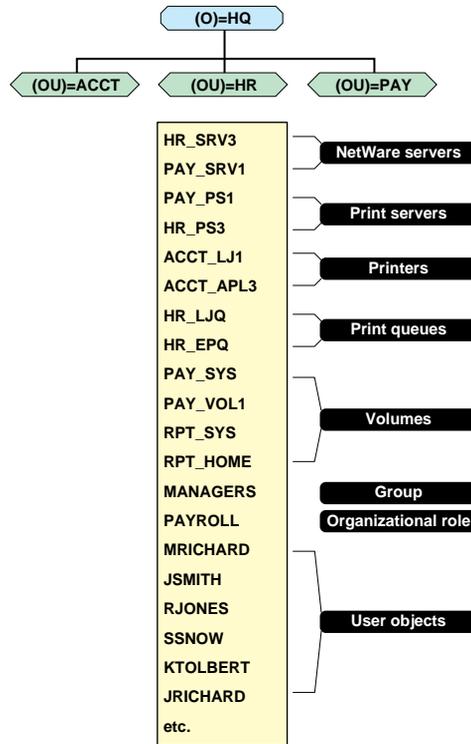
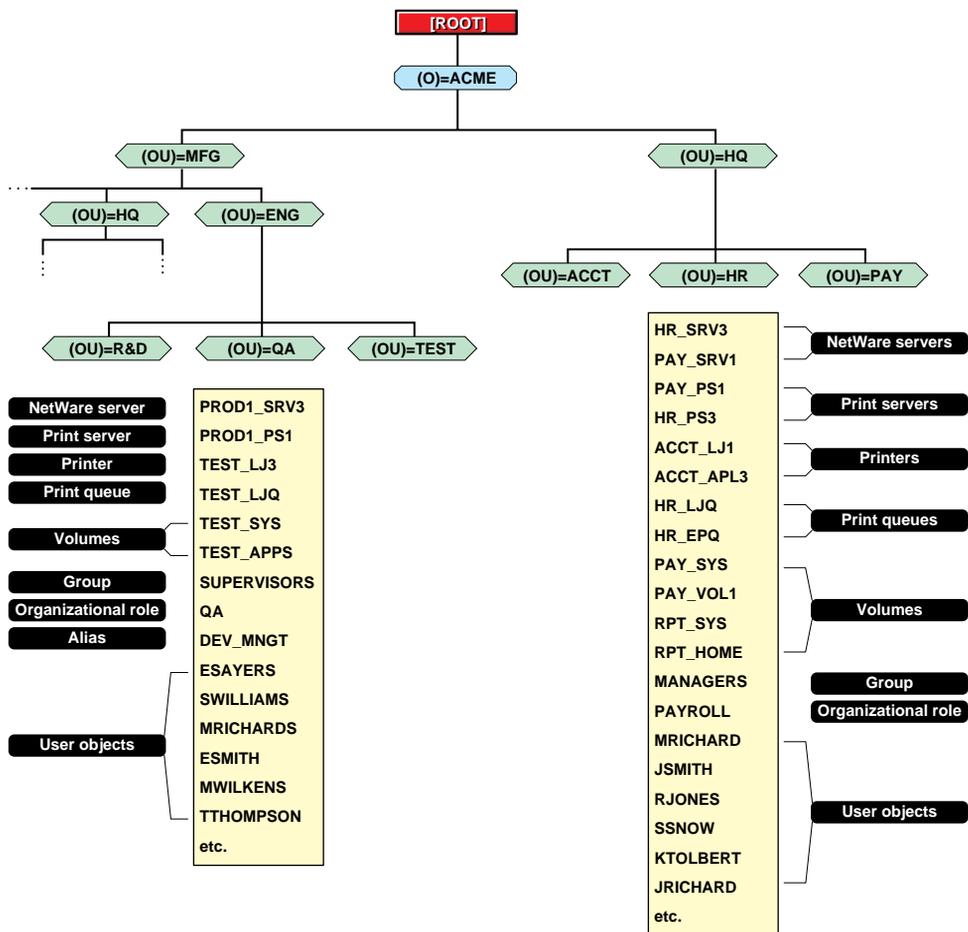


Figure 3-8
**Example of Leaf Objects in a
 Complex Tree Design**



Placing Shared Resources

Place shared leaf objects in the lowest container level that incorporates all the objects which need access to it.

For example, if a printer is used by two different departments which have separate Organizational Unit containers, place the Printer object a level above the two Organizational Unit containers for those departments.

Placing Server Objects

NetWare Server objects must be in or under the container that represents the physical location of the NetWare 4 server.

Place NetWare Server objects in the same container that contains the objects that access the server. For example, all User objects, services, print queues, and other leaf objects in the tree that attach to or are stored on a given server should reside in the same container.

You should also determine where to place central group servers such as e-mail and public servers. When considering where to place common servers, consider the amount of network traffic that will be generated on the wire and where users will find the server easily. Use Alias objects for these common servers to allow users easier access to them.

Designing for environment-wide aliases should be done when determining where to place common servers and resources.

Placing Objects for Mobile Users

Place Directory Map objects consistently throughout the tree to allow users to map a drive easily to a local server to access applications from any point in the Directory tree. This approach requires that some servers in the Directory tree maintain identical file system structures.

Place an Alias object of servers that mobile users commonly access close to the [Root] of the tree to make the authentication process faster.

Placing Objects for Global Resources

If an environment has many globally shared resources, design the tree to accommodate those resources. The following actions make the network more usable:

- ◆ When a user authenticates to the network, the profiles and scripts for that user are executed. If any of those are not kept locally, the login process retrieves those profiles and scripts across the LAN or WAN.

Keep the profiles and scripts either on the same server that the user's home directory is located on or relatively close to the user

across fast LAN or WAN links. This will decrease the time for the user during the authentication process.

- ◆ When a global application that needs access to the entire tree such as e-mail or calendars relies upon the Directory database, build a branch of the tree that contains Alias objects to the real objects.

This could be a single container just off of the [Root] object or a hierarchial subtree built directly below the [Root] object that contains an Alias object for every object that might be needed for the application.

Summary

After you have considered all the factors that will affect your tree design, lay out the final draft of your tree with a modeling or drawing application or diagram it on paper.

Remember that Novell Directory Services supports a high level of design flexibility. If you later want to change the layout, you can do it easily with the NetWare 4 utilities.

Evaluation

Once you have completed your design, review the following questions to evaluate the efficiency of your tree design:

- ◆ What logical structure of the Directory tree makes the most sense for the physical layout of your network?
- ◆ What organizational structure of the Directory tree makes the most sense for your network resources?
- ◆ How can resources be distributed across the network to reduce line traffic but still provide easy access for workgroups, users, and applications?
- ◆ How do you want the Directory database to be partitioned, and where do you want to store replicas of those partitions?
- ◆ How will your tree design affect the length and format of object names?

- ◆ How does your tree design support designing other services such as printing and security?
- ◆ How easily can this tree merge with other trees?
- ◆ Does this tree design allow for future growth?
- ◆ Does this tree provide efficient accessibility and navigation for local, global, and mobile users?

Defaults

The default settings for Directory objects are as follows:

Default Objects after NetWare 4 Installation	Default Objects after NetWare 4 Upgrade
NetWare Server object for the server on which NetWare 4 was installed.	NetWare Server object for the server you upgraded to NetWare 4.
Volume object SYS.	Volume object SYS for volume SYS: on the server you upgraded.
Volume objects for any other volumes besides SYS: that you created during installation.	Volume objects for every other volume besides SYS: on the server you upgraded.
User object ADMIN.	User object ADMIN.
When User object ADMIN is first created, by default it is placed in the Organization container object. This may not be the same context in which you installed the server.	When User object ADMIN is first created, by default it is placed in the Organization container object. This may not be the same context in which you installed the server. All other objects that were on the server's bindery are placed in the same container as the server that was upgraded.

Note: The upgrade utility converts most existing bindery objects to NDS objects.

Where to Go from Here

To	Go to
Determine the strategy you will use to scale and distribute the Directory database to servers throughout the network	Chapter 4, "Determining a Partition and Replication Strategy," on page 67
Plan the approach you will use to maintain a consistent network time for servers and clients on the network	Chapter 5, "Planning the Time Synchronization Strategy," on page 89
Create an accessibility plan for determining how network resources are accessed and used on the network	Chapter 6, "Creating an Accessibility Plan," on page 107

Determining a Partition and Replication Strategy

This chapter describes how to determine the partition strategy on your network. The following topics are discussed:

- ◆ “Forming Partition Boundaries” on page 70
- ◆ “Determining an Efficient Replica Placement Strategy” on page 75

If you have a single server environment, you do not need to determine a partition strategy. Continue with Chapter 5, “Planning the Time Synchronization Strategy,” on page 89.

Multiserver networks that meet the following conditions should accept all partition and replica defaults provided in NetWare[®] 4[™]:

- ◆ No WAN links
- ◆ No more than 15 servers
- ◆ Fewer than 1,000 objects

See “Defaults” on page 86 for more information.

Introduction

Directory objects and their attributes exist in a database that is maintained and managed by Novell[®] Directory Services[™] (NDS[™]). NDS distributes information about each Directory object to servers throughout the network.

NDS does this by dividing the database into logical segments of the Directory tree, and then copying each logical segment to a group of servers on the network. This logical segmenting is referred to as *partitioning*. The process of copying each logical segment to servers is referred to as *replication*.

Although partitioning and replicating the database means that not all of the Directory database information exists on each server throughout the network, links are established between servers to allow access to all of the database information. This does not mean, however, that every server on the network holds partition replicas. NetWare 4 utilities manage and monitor the partitioning process and the links between servers.

When determining the partition strategy, it is important to remember that partitioning is simply the logical segmenting of the Directory database, whereas replication is the physical placement of a partition on servers throughout the network. This means that you should consider not only the Directory tree structure but the physical network infrastructure.

You should also determine a partition strategy that provides the most effective placing of replicas without high management overhead or increased network traffic.

Objectives

You should determine a partition strategy for your network by forming partition boundaries and effectively placing replicas.

Use resource maps, location maps, LAN and WAN topology maps, the Directory tree structure, and partitioning guidelines to determine the partition boundaries and what types of replicas to store on which servers in the network.

Prerequisites

- You should have copies of the following planning documents:
 - ◆ WAN topology maps
 - ◆ Location maps
 - ◆ Resource maps
 - ◆ Directory tree structure
 - ◆ Partitioning guidelines
- You should have completed your Directory tree design. See Chapter 3, “Designing the Directory Tree Structure,” on page 25 for more information.

Determining Partition Requirements

Partitions should only be created to

- ◆ Reduce the dependency on a single network server
- ◆ Reduce the size of the database for any single sever
- ◆ Place resources near users for performance improvements
- ◆ Reduce network wire traffic caused by user access
- ◆ Reduce network traffic due to the synchronization process
- ◆ Decrease the administrative overhead

Avoid partitioning if it

- ◆ Causes administrative overhead
- ◆ Creates additional subordinate references for the child and parent partitions
- ◆ Increases network complexity
- ◆ Increases the time needed to navigate the tree
- ◆ Produces minor increases in network wire traffic

You need to determine the best balance of advantages and disadvantages before creating a partition strategy for your tree design.

Forming Partition Boundaries

Forming effective partition boundaries enables you to scale and optimize the Directory database on your network. To do this, you should understand some basic characteristics of partitions.

Considering Partition Characteristics

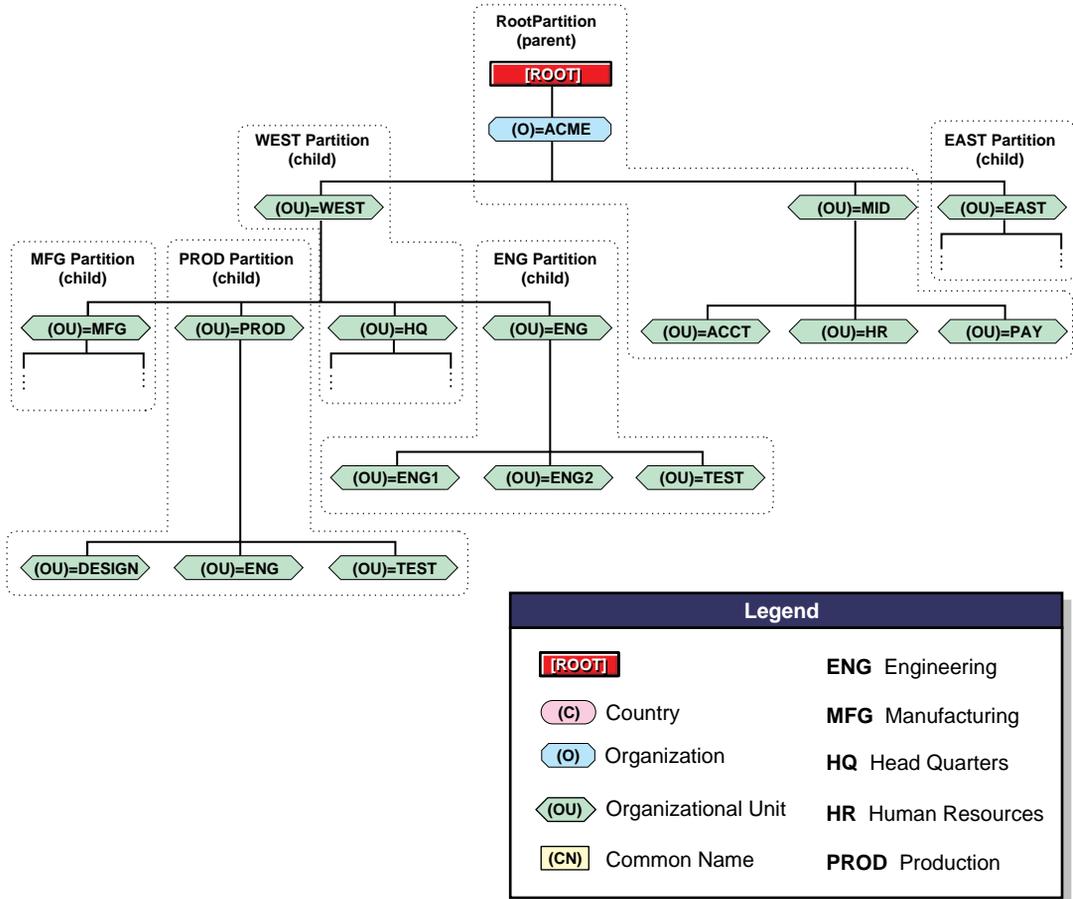
A partition is a *subtree* or *branch* of the Directory tree. Each partition is named according to the *partition root object*. This is the highest container object in the partition. This container is also referred to as the *partition root entry*.

The [Root] object is always included in the first partition created, which is known as the *[Root] partition* .

When a partition is subordinate to another in the Directory tree, it is referred to as a *child partition*. The partition above it is referred to as the *parent partition* .

The following illustration shows a parent partition in relation to its child partition in a Directory tree.

Figure 4-1
Parent and Child Partitions



Partitions must follow a strict set of rules. These rules are as follows:

- ◆ A partition can contain only Directory objects and related data. It cannot include any information about the file system directories and files.
- ◆ A Directory object can exist in only one partition.
- ◆ Partitions can be stored only on NetWare 4 servers.
- ◆ Partitions cannot overlap each other.
- ◆ Partitions must contain a connected subtree.

Planning the Partition Layout

When the first server of a Directory tree is installed, a partition is created at the [Root] object. This partition is referred to as the [Root] partition and contains the entire Directory tree at that time.

The [Root] partition is the only partition that the installation process creates. All other partitions must be created manually.

The design of your partitions should resemble a triangle with a small number of partitions at the top level of the tree and more partitions as you move toward the bottom. Such a design will create fewer subordinate references than a tree that has more partitions at the top than at the bottom.

This triangular design can be accomplished if you always create the partitions relatively close to the leaf objects (particularly the users). An exception is the [Root] partition.

Designing Partition Boundaries

In most cases, you should design partition boundaries around the physical layout of your network infrastructure. This coincides very well with the approach used in designing the upper and lower layers of the Directory tree.

As a rule, if your tree design is structured correctly, your partition strategy will be easy to implement and maintain. Consider the following guidelines when partitioning the Directory tree:

- ◆ Design upper layer partitions around Organizational Unit boundaries that represent location or organizational structures.
- ◆ Plan the partitioning so that partitions contain objects from a single campus. Do not allow partitions to span slow WAN links.

A single campus consists of a single location or multiple locations connected by high speed (T1 or better) WAN links.

Subordinate reference replicas will exist across WAN links if no partitions are spanned across WAN links. Ensure that your strategy creates a balance between maintaining local partitions and reducing the number of subordinate reference replicas that are created.

Also consider NDS backup issues when creating local partitions. If you plan to regularly back up the entire tree from a central location, consider the amount of time it may require if information needs to cross WAN links.

- ◆ After installing the first NetWare 4 server, use NDS Manager or PARTMGR to create partitions before installing subsequent network servers.

Hint: One approach that has been used successfully and reduces the amount of manual partition placement is to partition the tree first and then add additional servers into the tree for each partition.

- ◆ Maintain small partitions.
- ◆ Place partitions near the Directory objects (particularly User objects) that use the resource contained within the partition.
- ◆ Group servers on the same cabling segment into the same partition if you split partitions because of any the following conditions:
 - ◆ WAN links exist
 - ◆ More than five servers exist in a partition
 - ◆ More than 1,000 objects exist in a partition

Designing Partitions for Upper Layers

The first layer of partitions after the [Root] partition is defined according to the physical locations within your organization and the network infrastructure. For example, if your organization has multiple campus sites located across the country, your tree design should reflect this geographical separation. Upper layer partitions should be established at each of these individual sites.

If your organization is contained at a single site, your upper layer partitions should reflect the organizational structure of the upper layers. Partitioning at a single campus site, however, is more dependent on the number of objects within a branch of containers than the physical infrastructure. This is because no slow WAN links exist.

If WAN bandwidth is an issue, create smaller partitions and replicate them in fewer locations with fewer subordinate partitions. This produces less network traffic when synchronizing across WAN links.

Designing Lower Layer Partitions

Lower layer partitions are defined according to the organizational divisions within the organization and how they are reflected in the tree design. The Directory tree design should identify division, department, and workgroup structures within an organization. Partitioning should be patterned along these same lines.

Determining the Size and Number of Partitions

The size of the partitions can significantly affect the synchronization and responsiveness of the system. As you plan partitions, you should balance the factors concerned with its size.

You should consider the following:

- ◆ Partitions that contain more than 1,000 objects will cause long delays in synchronization and high levels of network traffic depending on the network hardware used.
- ◆ Partitions that contain fewer than 50 objects may cause more management overhead than their worth.
- ◆ A tree design that contains many slow WAN links may require additional partitions.
- ◆ Your administrative model may require more partitions at the lower level to better define areas of management and control.
- ◆ The amount of network traffic your network can support will affect the number of partitions.
- ◆ The static nature of objects in the tree may determine the number of partitions. If your network is rapidly growing, you should allow for future growth by maintaining relatively small partitions. Nevertheless, splitting or joining partitions is a simple task.

Note: If you maintain large partitions, performing partition operations will be time consuming if network throughput is slow and a large number of objects are being affected.

Determining an Efficient Replica Placement Strategy

Determining the most effective replica placement strategy allows you to optimize the Directory database for fault tolerance, accessibility, and navigation. To do this, you should understand some basic characteristics of replicas.

Considering Replica Characteristics

A replica is basically a physical copy of a partition. An unlimited number of replicas for each partition can be stored on any NetWare 4 server on the network. Also, a single NetWare 4 server can store multiple replicas if they are generated from unique partitions.

Basic Functions of Replicas

Replicas provide the following three functions to the network:

- ◆ *Directory fault tolerance.* If a disk crashes or a server goes down, a replica on a server in another location can still authenticate users to the network and provide information on objects in the disabled server's replica.

With the same information distributed on several servers, clients are not dependent on any single server for network authentication or to provide services.

Important: If your network has only one server, you will most likely maintain only a single copy of the [Root] partition. This is because your network infrastructure doesn't require it.

Under this scenario, the only means of providing some level of fault tolerance is by maintaining a up-to-date backup copy of the Directory database. Make sure that your backup software can back up NDS.

Also, partition replication does not provide fault tolerance for the file system. Only information about Directory objects is replicated. To provide fault tolerance for your files, you must mirror or duplex your hard disks and enable the Transaction Tracking System™ (TTS™) feature.

- ◆ *Increased access performance.* Distributing specific Directory information to servers that are physically near workgroups or users who frequently access the information greatly increase the performance of authentications, modifications, and searches. This

is because information is retrieved from the nearest available server containing the specified information.

Access to information that exists in partitions across WAN links is improved and WAN traffic is decreased, depending on the amount of synchronization that is required.

- ◆ *Tree walking.* Finding specific information in the Directory tree is referred to as *tree walking*, or *name resolution*. This technology allows clients to locate Directory information that doesn't exist within the container their User object resides in. NDS performs the name resolution process to locate the information.

Every replica maintains references (pointers) to servers that store those replicas that are subordinate or superior to its relative position in the Directory tree. NDS uses these pointers to walk up or down the tree to find the information that is requested.

Placing replicas of frequently accessed information on local servers increases the speed at which names are resolved.

Identifying the Four Types of Replicas

There are four types of replicas.

- ◆ *Master replica.* A master replica is a writable replica that contains all object information for the partition. All partition operations (create, merge, move, delete) occur from the master replica of the given partition.

Only one master replica can be defined for each partition.

- ◆ *Read/write replica.* A read/write replica contains the same object information as the master replica. It allows modifications (writes) when a master replica of the given partition is already defined somewhere else.

There can be any number of read/write replicas.

Client workstations can update master and read/write replicas only.

- ◆ *Read-only replica.* A read-only replica contains the same object information as the partition, but the information can only be read. It is used where reading the partition is required, but writes to the partition should not occur.

Note: A read-only replica cannot be used on a server where bindery services is required because bindery services requires a writable replica. When bindery services is set, use either a master or read/write replica.

The default setting in the NetWare 4 installation program copies a read/write replica of the partition that a bindery server is being upgraded to.

- ◆ *Subordinate reference replica.* A subordinate reference replica cannot be modified by any user. It is automatically placed on a server by NDS if the parent partition has a master, read/write, or read-only replica on the server and the child partition does not exist on the server.

If you add a read/write or read-only replica of the child partition to the server, the subordinate reference replica is removed.

Network resources are held in the master, read/write and read-only replicas. Read-only replicas have limited scope and are not typically implemented. Subordinate reference replicas are automatically allocated and created, and tie the partitions of the tree together.

Identifying How Replicas Are Updated and Synchronized

When changes are made to objects within a partition, those changes are automatically sent to all other replicas of that partition. This ensures that the global Directory database remains consistent. Only changes are sent to other replicas. For example, if a user changes a phone number, only the new phone number is sent, not the entire User object.

The master replica participates in the partition synchronization process by exchanging updates with other replicas but does not control this process. Similarly, each read/write replica synchronizes with the other replicas of the partition. Read-only replicas also synchronize with other replicas, but they receive updates only from other servers.

An NDS database is a loosely consistent database. As changes occur, all replicas of a partition do not always contain exactly the same information at every instant. In fact, the contents of the replicas most likely vary slightly at any given time. However, these replicas

eventually converge to a consistent state once the changes are distributed to all replicas.

Some changes are sent immediately to other replicas, such as changes to a user's password. Less critical changes, such as a user's last login time, are collected locally for a short period of time before being sent out to the network.

For this synchronization to occur, each replica must be contacted. When a partition is created, some additional attributes are added to the container object that is acting as the partition root object. These properties are used to manage the synchronization of data between replicas of the partition.

Three attributes should be considered when considering synchronization issues:

Attribute	Description
Replica Pointer	Each partition maintains a record of the location of each of its replicas. The locations are stored in the partition's replica property, with one property entry for each replica of the partition. The collection of replica pointers of a partition forms a list of the partition's replicas, sometimes called a <i>replica ring</i> or <i>replica list</i> .
Synchronized Up To	Each replica in the replica list for a partition maintains a list of time stamps. The server holding a particular replica uses this attribute to determine the synchronization state of the replica it's holding in comparison to other replicas in the replica list. This attribute is sometimes called a <i>vector of time stamps</i> .
Inherited Access Control List (ACL)	The partition root object is responsible for determining the access rights inherited from the [Root] object down to itself. This attribute is primarily a summary of the ACL for all objects within its partition.

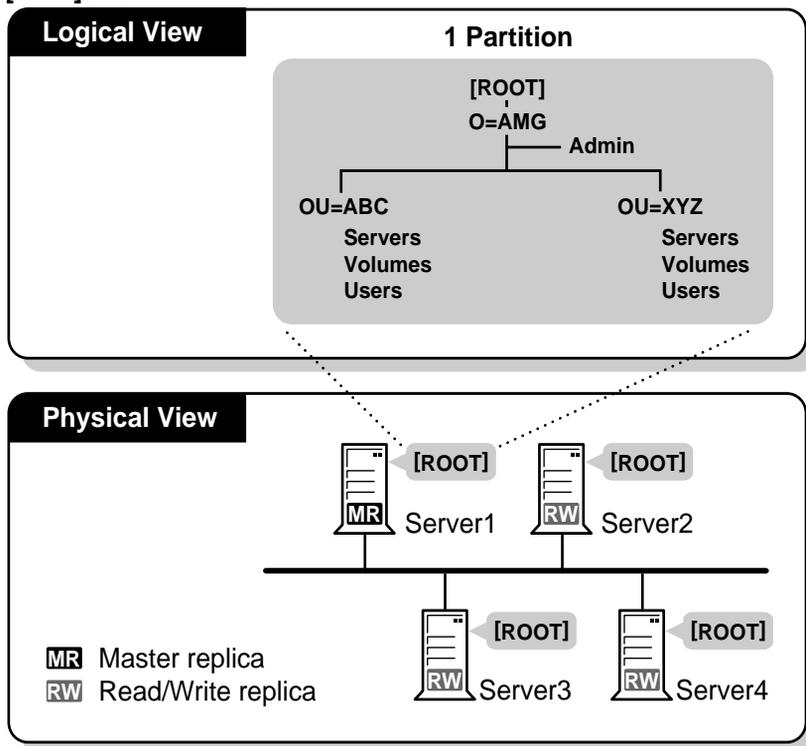
Planning Replica Placement

When the first server of a Directory tree is installed, the first replica of the [Root] partition is placed on that server. This first server contains the master replica of the [Root] partition.

There are no special rights or considerations for this server. The replica of [Root] can be removed at any time or changed to a read/write replica once other servers are in the tree.

The following figure illustrates how the [Root] partition can be replicated on your network.

Figure 4-2
Example of [Root] Partition Placement



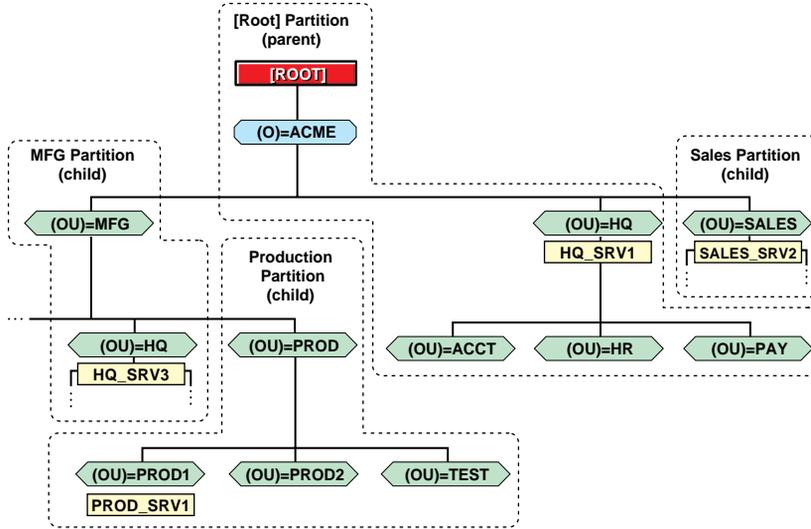
When you install additional servers in the tree, follow two simple rules to determine if a replica should be placed on the server.

- ◆ If you are upgrading the server from Netware 3™ to Netware 4, a replica is placed on that server to support bindery services for up to three servers in a given partition.
- ◆ If fewer than three replicas of a given partition exist in the tree, a replica is placed on the server to provide for fault tolerance and disaster recovery.

In all other cases, if a replica is desired on a server, the replica will have to be added manually.

The following figure illustrates how replicas can be placed on server in the Directory tree.

Figure 4-3
Example of Replica Placement



		Partitions				
		[Root]	PROD	Sales	MFG	
Servers	1	PROD_SRV1	R/W	MR	SR	SR
	2	SALES_SRV2	R/W	SR	MR	R/W
	1	HQ_SRV3	R/W	SR	R/W	MR
	1	HQ_SRV1	MR	R/W	R/W	R/W

Legend		
[ROOT]	MR Master replica	S R Single Reference time server
(C) Country	R/W Read/Write replica	R Reference time server
(O) Organization	RO Read-Only replica	1 Primary time server
(OU) Organizational Unit	SR Subordinate Reference replica	2 Secondary server
(CN) Common Name		

Designing Replica Placement

In most cases, you should design replica placement around the principles of fault tolerance, accessibility, and navigation. Consider the following guidelines:

- ◆ Create a minimum of three replicas of each partition.
- ◆ Replicate the [Root] partition at least three times (this partition is required to maintain the integrity of the tree). The server installation program automatically creates one replica of the [Root] partition. Other replicas must be created manually.
- ◆ Maintain performance by placing a replica containing information on a server that the users access (located within a single campus).
- ◆ Create replicas as needed for bindery services.
- ◆ Consider the NDS access performance goals, such as tree walking and keeping partitions and replicas close to the users.

Placing Replicas for Fault Tolerance

Create a minimum of three replicas of each partition. You may want to create more, depending on network topology or performance.

Place at least one copy of each replica off site, in another building or location. You may want more locations, depending on the disaster recovery plan being used by the organization. You can rebuild most of the network by using partition replicas.

Ensure that subordinate reference replicas are not used for fault tolerance. Subordinate reference replicas can be used to restore the tree structure, but not leaf objects.

Placing Replicas for Accessibility

Place replicas in the location of highest access. This means that if groups of users in two separate containers need access to the same object within another partition boundary, you should place the replica on a server that exists in the container one level above the two containers holding the groups.

Placing Replicas for Navigation

Place replicas on servers that contain objects for users and resources that are physically near the server. This allows for faster response to users' requests for login authentication and access to network resources.

Place replicas on the server that stores the user's HOME directories that access it. Also, store replicas of Directory objects that exist on opposite sides of a WAN link on the local servers that users access.

Placing Replicas for Administration

Partition operations require that a master replica of each partition be accessible. You should ensure that master replicas exist on servers that are located physically near the administrator of the partition.

Placing Replicas for Bindery Services

Bindery services is enabled at installation or is automatically enabled if you are upgrading a server from NetWare 2 or NetWare 3.

If bindery services is enabled, the server receives a read/write replica of up to three partitions that contain a container object in its bindery context.

Consider the following guidelines for bindery services support:

1. Identify all containers that hold bindery resources and record their context. (This is referred to as *bindery context*.)
2. Identify the partitions that hold the containers with bindery resources.
3. Place a read/write replica of those partitions on the server.

Placing Replicas for Servers

Place as few replicas on a server as possible.

Place replicas on the best servers in your network. Slow servers can affect the replica synchronization for all servers within the replica ring. (The replica ring refers to the group of servers that hold replicas of the same partition.)

Creating a Replication Matrix

You should create a replication matrix for your network. Use the following table to organize the physical placement of the replicas for each partition. We recommend that each partition have three replicas for fault tolerance, but you may need more replicas depending on user access. See Appendix C, “Template Examples,” on page 251 for more information.

Summary

Partitioning and replica placement allow you to scale your Directory tree to meet your organization's needs. This process can be as simple as accepting all defaults or as complex as designing a partitioning and replica placement matrix to define many locations for your partitions and replicas.

The following table list reviews some of the basic guidelines to follow when partitioning your Directory:

Condition	Guideline
Location	<ul style="list-style-type: none">◆ In a network with WAN links, partitions should not span multiple locations◆ Partition locally around the servers (keep physically distant servers in separate partitions)◆ Place fewer partitions at the top of the tree with more at the bottom
Size	<ul style="list-style-type: none">◆ Keep partition sizes small◆ The [Root] partition should remain small◆ Typically, a partition should have fewer than 1,000 objects, and no more than 3,500◆ Typically, a partition should have fewer than ten to fifteen subordinate partitions, and no more than forty

The following table list reviews some of the basic guidelines to follow when placing replicas in your Directory:

Condition	Guideline
Placement	<ul style="list-style-type: none">◆ Replicate locally, not across a WAN link (Replicas across a WAN link have to send/receive NDS synchronization information, which can slow down network traffic across a WAN link)◆ If possible, place master replicas physically close to the master of parent and child partitions
Number	<ul style="list-style-type: none">◆ Always keep two or three replicas per partition, and no more than ten◆ Never store more than ten replicas on a server

Evaluation

Once you have completed your partition strategy, review the following questions to evaluate the efficiency of your strategy:

- ◆ Are the upper layer partitions formed around Organizational Unit container boundaries that represent location or organizational structures?
- ◆ Do partitions contain only objects from single campus or span high speed (T1 or better) WAN links?
- ◆ Do the partitions contain fewer than 1,000 objects?
- ◆ Are partitions located near the objects (particularly User objects) that use the resources contained within the partition?
- ◆ Are servers physically located on the same cabling segment grouped into the same partition?
- ◆ Is there a minimum of three replicas of each partition?

- ◆ Do replicas exist to support bindery services?
- ◆ Are replicas stored on servers to meet your access performance goals, such as tree walking and keeping partitions and replicas close to the users?

Defaults

The defaults for partitioning and replication are as follows:

Replica Type	Default
Master	<p>The first NetWare 4 server in the network receives the master replica of the [Root] partition.</p> <p>The first server in a partition receives the master replica of that partition.</p>
Read/write	<p>New servers</p> <p>The second and third servers in a partition receive read/write replicas of that partition. Subsequent servers receive no replicas unless bindery services is enabled at installation.</p> <p>If bindery services is enabled on the new server, the server receives a read/write replica of up to three partitions that have a container object in its bindery context.</p> <p>Upgraded server</p> <p>A server upgraded from NetWare 3 receives a read/write replica of up to three partitions that have a container object in its bindery context.</p> <p>Note: Bindery services is enabled at installation or is automatically enabled if you are upgrading a server from NetWare 3.</p>
Read-only	None.
Subordinate reference	Subordinate reference replicas are automatically allocated and created, and tie the partitions of the tree together.

Where to Go from Here

To	Go to
Plan the approach you will use to maintain a consistent network time for servers and clients	Chapter 5, "Planning the Time Synchronization Strategy," on page 89
Create an accessibility plan for determining how network resources are accessed and used	Chapter 6, "Creating an Accessibility Plan," on page 107
Create an application management strategy for network applications to improve and efficiently manage network applications	Chapter 8, "Designing an Application Management Strategy," on page 159
Create a migration strategy for servers and workstations from a previous version of NetWare or other network operating system	Chapter 9, "Developing a Migration Strategy," on page 173
Create an implementation schedule	Chapter 10, "Creating an Implementation Schedule," on page 197

Planning the Time Synchronization Strategy

This chapter describes how to plan a time synchronization strategy. The following topics are discussed:

- ◆ “Determining an Efficient Time Synchronization Method” on page 91
- ◆ “Identifying an Efficient Time Source and Configuration” on page 95

If you have a single server environment, you do not need to plan for time synchronization. Continue to the next procedure. See Chapter 6, “Creating an Accessibility Plan,” on page 107 for more information.

Multiserver networks that meet the following conditions should accept all time synchronization defaults provided in NetWare[®] 4[™]:

- ◆ No WAN links exist
- ◆ Fewer than thirty servers exist
- ◆ Single time zone exists

See “Defaults” on page 104 for more information. You might also want to review information about using a combination of internal and external time synchronization methods. See “Using a Combination of Time Synchronization Methods” on page 94 for more information.

Introduction

Directory objects exist in a database that is maintained and managed by NetWare Directory Services™ (NDS™). The database can exist on a single server or can be partitioned and distributed as replicas on other servers. NDS ensures that when changes are made to objects in a partition, the changes are made to all replicas of that partition in the proper order in which they were performed.

If multiple events exist for a particular object within a partition, NDS checks that the events are made to the object in their proper sequence. An example of this would be one administrator renaming an object and another moving the object. If these events occurred out of sequence, the object might not be renamed because it would not exist in the original tree context.

To ensure that events occur in their proper sequence, NDS records the time of each event relative to a common time source. This time record is referred to as a *time stamp*. NDS uses the time stamp to ensure that when it modifies the database, the particular event occurs in the time and order that it was intended. NDS also uses time stamps to record real world time values for the network and set expiration dates.

In a single server environment, the server's internal clock is adequate to maintain a common and consistent time source for the network.

In a multiple server environment, NetWare 4 includes functionality to maintain a common time for all NetWare 4 servers in the network. It is referred to as *time synchronization*.

The common time source necessary for tracking the proper sequence of events is provided through time synchronization across all NetWare 4 servers. You should plan your time synchronization strategy to ensure that a common time source is maintained efficiently and accurately.

Note: The standard format used for times and time offsets is [+|-] HH:MM:SS. In practice, only the significant portion of the time need be specified (+7:0:0 is the same as 7).

In addition, the examples used in this chapter show the colon as a time separator. The actual character used as the time separator is determined by the country information for each server. In most instances in NetWare 4, input and output routines dealing with dates and times are language enabled to use the locally preferred format and characters.

Objectives

The objective in planning the time synchronization strategy is to determine an efficient time synchronization method to use, and then identify the best way to set up and manage that method across the network.

Use resource maps, location maps, LAN and WAN topology maps, and the tree design document to determine which time servers to use and what communication format to use between the network servers.

Prerequisites

- ❑ You should have copies of the following planning documents:
 - ◆ Resource maps
 - ◆ Location maps
 - ◆ LAN and WAN topology maps
 - ◆ Directory tree design

Determining an Efficient Time Synchronization Method

NetWare 4 needs to maintain a correct system time (real world time) for keeping file date and time stamps correctly, for auditing and logging, and to manage user's login time restrictions. It is also important to maintain a common time for the entire network system of servers.

Time synchronization provides mechanisms to adjust and compensate for the rate of the operating system software clock. In addition, it also provides support for Universal Time Coordinated (UTC) time, the worldwide time standard coordinated to the zero meridian (0° of longitude, also known as (GMT) Greenwich Mean Time).

This is done by designating one or a group of servers to act as the time source for all other servers and client workstations in the network. All other servers act as secondary time servers, receiving UTC values from the time source. This relationship ensures that any drifts between time settings of different server's internal clocks are corrected and common time is maintained.

Establishing a common time source for the network can be done using one or a combination of the following two methods:

- ◆ Internal time synchronization
- ◆ External time synchronization

Using Internal Time Synchronization Methods

NetWare 4 provides you with the functionality to maintain the same UTC time on all servers. This is done through a server utility named TIMESYNC.NLM. TIMESYNC allows you to set up different types of time source servers that provide time to other servers and clients.

Using TIMESYNC

All NetWare 4 servers automatically load TIMESYNC, which manages the updating of each server's Universal Coordinated Time (UTC) relative to the UTC of the network. Time synchronization is active whenever TIMESYNC is loaded.

TIMESYNC determines if its internal clock coordinates with that of the time provider. This is done by determining if the internal clock time is synchronized within a specific time radius of the value received from the time provider. If the synchronization radius is within the set amount of time, the server sends a time synchronization flag to indicate that synchronization is established.

The synchronization radius is set with TIMESYNC parameters in the SERVMAN or SET utilities. The default setting is 2000 milliseconds, or roughly 2 seconds.

Identifying NetWare Time Servers

The TIMESYNC utility allows you to set up four types of time servers in NetWare 4. See the following table for a listing and description.

Table 5-1
Time Server Types

Server Type	Function	Caution	Description
Secondary (Default)	Receives time from a time source server and provides time to client workstations.	You can have a Secondary time server contact another Secondary time server to obtain the correct time. However, if the intermediate Secondary time server is unavailable, servers that contact it for the correct time might be too many hops away from a time source server to get synchronized.	Attempts to stay synchronized with one time source. Does not participate in voting. Most servers will be secondary servers.
Primary time server	Polls and votes with other Primary time servers to determine time, and provides time to Secondary time servers and client workstations. Use with Reference time servers to pass time to Secondary time servers and client workstations.	Must be able to contact at least one other Primary time server or a Reference time server.	Polls all other time sources to determine correct network time and compensate for clock errors Sets synchronization status based on its deviation from calculated network time, without regard to status of other time sources polled.
Single Reference time server	Provides time to Secondary time servers and client workstations. Typically used for smaller LANs.	All servers must be able to contact the Single Reference time server. No Primary or Reference time servers can be on the network.	Functions the same as Reference time server, however it does not synchronize with any other server.
Reference time server	Receives time from an external time source and provides time to Primary and Secondary time servers. Use when it is important to have a central point of control for time on the network.	Typically, only one Reference time server is installed on a network. If there is more than one Reference time server, each must be synchronized with the same external time source.	Functions the same as a Primary time server, except that it does not adjust its internal clock. Provides a central point of time control for the entire network.

Each time-synchronized server performs three fundamental functions:

- ◆ Provides UTC time to any NLM or client workstation that requests it
- ◆ Provides status information indicating whether the UTC time is synchronized
- ◆ Adjusts its internal clock rate to correct for drift and maintain UTC synchronization

See “Identifying an Efficient Time Source and Configuration” on page 95 for more information.

Using External Time Synchronization Methods

Common time can be established by connecting all network servers to the same external time source. Depending on the cost of the source or hardware setup, this can be an effective method for maintaining common time on your network.

Some examples of external time source are radio clocks, atomic clocks, or Internet time.

Using a Combination of Time Synchronization Methods

Time synchronization services does not attempt to provide correct time of day to the servers. It can only keep the internal clocks of each server set to a common time.

You can ensure the accuracy of your network's time of day by simply checking the time provider's Universal Coordinated Time (UTC) value against a wristwatch, or by attaching the time provider to an external time source service through a modem, radio, or Internet time source.

Hint: Adjust the synchronization radius if the time provider servers are separated by WAN or satellite links that cause delays in packet transmissions.

Identifying an Efficient Time Source and Configuration

Identifying the best time source and configuration for your network depends on the following tasks:

- ◆ Identifying the time source to use
- ◆ Determining an efficient configuration
- ◆ Identifying the communication format

Identifying an Efficient Time Source

Identifying the time source type for your network is based on the method of synchronization you choose. See “Determining an Efficient Time Synchronization Method” on page 91 for more information.

Time source:for internal time synchronizationTime servers are divided into two types that act as time providers or time consumers. These servers function on primarily the same principles. These principles are:

- ◆ All servers provide time to any time provider, time consumer, or workstation.
- ◆ All servers are responsible for their own synchronization, meaning that they must decide if they're synchronized to the network's Universal Coordinated Time (UTC) or not, and report its synchronization status.
- ◆ All servers must adjust their internal clock rates to correct discrepancies and maintain time synchronization with other network servers.

Internal Time Sources

Time synchronization provided in NetWare 4 supports three kinds of time providers.

Time synchronization identifies all time consumers as Secondary time servers.

Secondary time servers obtain the time from a Single Reference, Primary, or Reference time server. They adjust their internal clocks to synchronize with the network time, and they provide the time to client workstations.

A Secondary time server doesn't participate in determining the correct network time.

External Time Sources

Time source:for external time synchronizationIf you use an external time synchronization method, then the time provider is identified as an external time source.

Modem and radio sources are most commonly used to synchronize the clocks on workstations and NetWare servers.

Note: A list of third-party time source services is maintained in NOVLIB forum on NetWare. The file is currently named TIMESG.TXT.

Use your planning documentation and documentation provided by the time source to identify setup and maintenance procedures.

Determining an Efficient Configuration

Time source:efficient configuration, determiningDetermining an efficient configuration of time synchronization depends on the physical layout of your network infrastructure. You should reference any planning documents related to the LAN and WAN topology of your network.

There are basically two efficient time server configurations. The two configurations are:

- ◆ Single Reference
- ◆ Time provider group

Select one of the time server configurations to model the time synchronization strategy of your network.

Using a Single Reference Time Server

Time source configuration: single reference time server, using NetWare 4 uses a Single Reference time server as its default configuration. The first server installed into a NetWare 4 network is automatically configured as a Single Reference time server. All other NetWare 4 servers installed into the network are configured as Secondary time servers.

The Single Reference time server configuration is adequate for networks with the following conditions:

- ◆ Fewer than thirty NetWare 4 servers
- ◆ No WAN links
- ◆ Single time zone

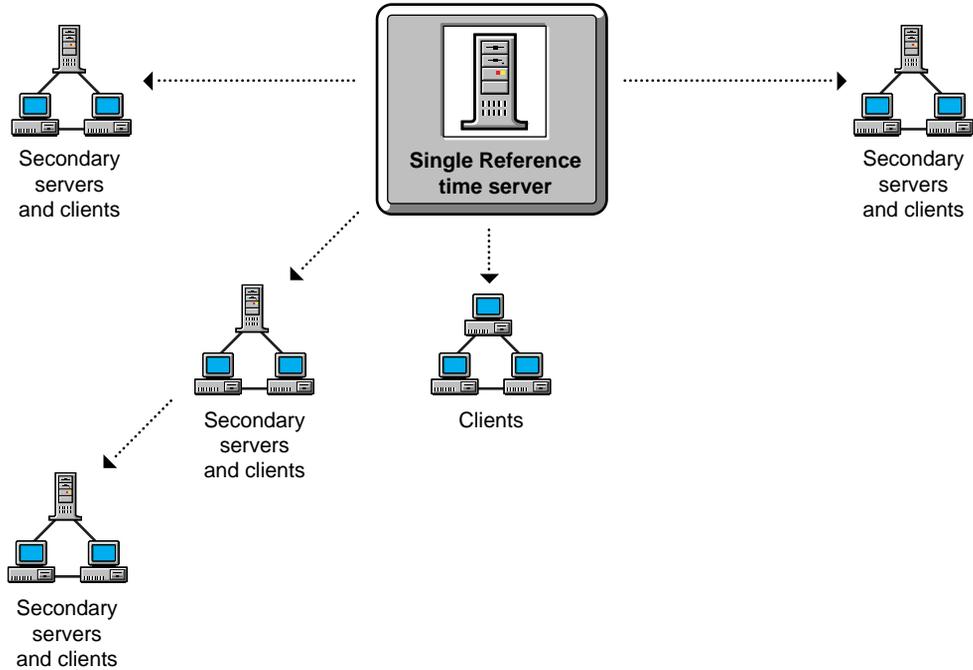
You should ensure that the Single Reference time server is centrally located and is equipped with an accurate internal clock. The Single Reference time server should be monitored on a daily basis to ensure proper time of day and time synchronization.

A limitation to a Single Reference time server configuration is the lack of fault tolerance if the time server loses its connection for an extended period of time. This may cause Secondary servers to fall out of sync with the network Universal Configured Time (UTC).

However, if the Single Reference time server loses its connection, you can designate one of the Secondary time servers as a temporary Single Reference time server until the original is reconnected.

The following figure illustrates a Single Reference time server providing time to Secondary time servers and to its own client workstations. The Secondary time servers, in turn, provide time to their own client workstations.

Figure 5-1
Single Reference Time Server



The Single Reference time server works on networks of any size, but the time synchronization configuration shown in Figure 5-1 is used primarily for small networks that don't include WAN links.

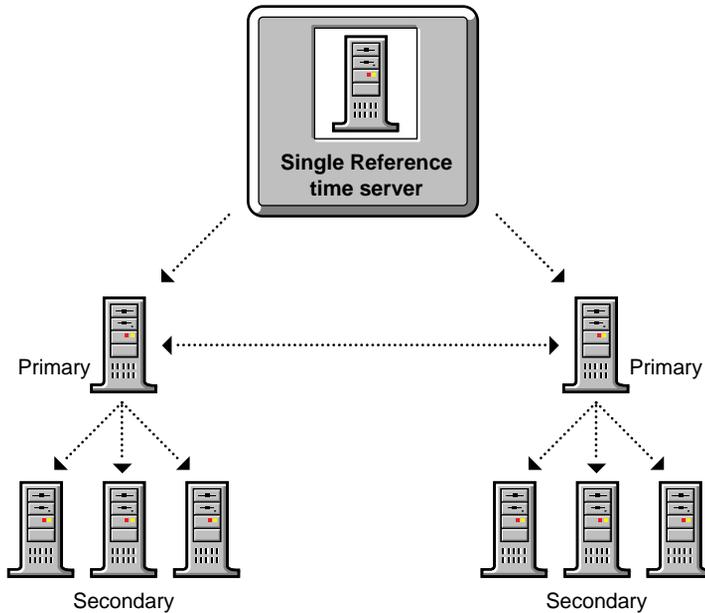
Important: If you use a Single Reference time server, avoid using Primary or Reference time servers in the same tree because the time references might conflict.

Using a Time Provider Group

Using a time provider group requires that at least one server is designated as a Reference time server and two servers are designated as Primary time servers. The Reference time server polls the Primary time servers within the time provider group to vote on the correct time.

The following figure illustrates a Single Reference time server providing time to primary time servers. Primary servers, in turn, provide time to Secondary time servers. Each of these servers can provide time to their own client workstations.

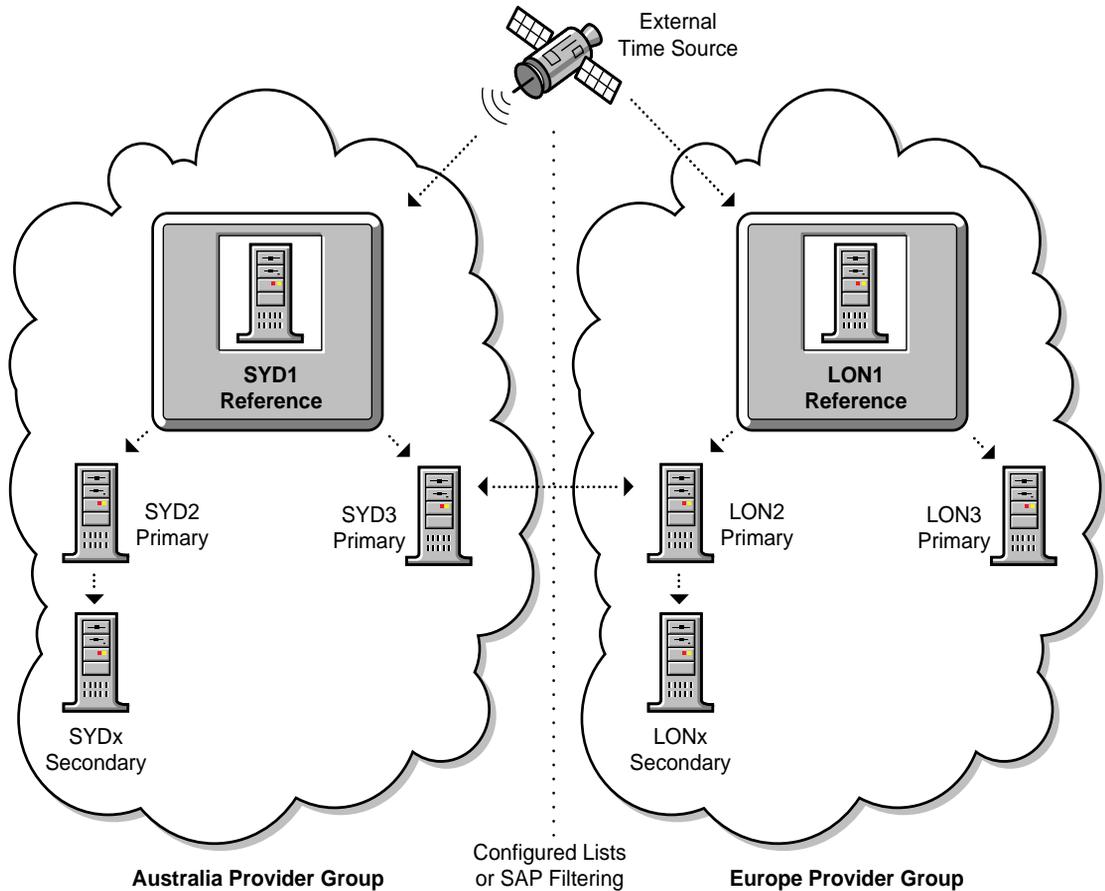
Figure 5-2
Time Provider Group with a Single Reference
Time Server



Usually, only one Reference time server is installed on a network. If you use more than one Reference time server, you must synchronize each Reference time server with the same external time source, such as a radio clock, atomic clock, or Internet time.

The following figure illustrates a network using an external time server to provide time to two individual Reference time servers.

Figure 5-3
Two Reference Time Servers Using an
External Time Source



This configuration requires that you modify the time parameters on the NetWare 4 servers in the time provider group.

The NetWare 4 server that is designated as the Reference time server should be placed in a central location. NetWare 4 servers designated as Primary time servers should be located either at the same central location as the Reference time server or used as locational time servers.

If you are configuring for a multiple campus network, you should distribute Primary time servers strategically across the WAN

infrastructure. This will reduce WAN traffic by providing a time source for Secondary time servers and client workstations at each location.

If your WAN infrastructure forces you to have more than seven Primary time servers in the time provider group, you should implement additional time provider groups as necessary. However, you should ensure that each Reference time server is synchronized to the same external time source. Reference time servers cannot synchronize time with other Reference time servers.

All other servers in the network should be designated as Secondary time servers.

Identifying the Communication Format

Time synchronization:communication format, identifyingTime providers and time consumers need to communicate in order to send and receive time. Time providers need to communicate with other providers in order to vote and negotiate the correct network Universal Coordinated Time.

Time source servers use one of two methods to find each other: SAP, or a custom configuration list.

Using SAP (Service Advertising Protocol)

Time synchronization:communication format, using Service Advertising Protocol (SAP)By default, Primary, Reference, and Single Reference time servers use the Server Advertising Protocol (SAP) to announce their presence on the network. Time consumers do not use SAP.

Primary and Reference time servers use the SAP information to determine which other servers to poll in order to determine the network time.

Secondary time servers use the SAP information to choose a time server to synchronize with.

An advantage of the SAP method is that it allows for quick installation without regard to the network layout. It also allows automatic reconfiguration if operating modes are changed or if new servers are added to the network.

The SAP method, however, generates additional network traffic.

The SAP method can also be disruptive in large network environments where test servers come and go, especially if the test server is configured as a time source (Single, Reference, or Primary time server).

Using a Custom Configuration List

Time synchronization:communication format, using custom configuration listCustom configuration of your time servers gives you more control over time synchronization, but it requires more planning to synchronize servers efficiently.

An advantage of creating a custom configuration list is that you maintain complete control of the time synchronization environment.

Also, custom configuration lists help eliminate nonessential network SAP traffic, as well as errors associated with accidental reconfiguration.

It is also possible to list the specific time source servers that a server should contact.

It is possible to specify that a server should not listen for SAP information from other time source servers and that it is not to advertise its presence using SAP.

The custom configuration does require additional time for planning and installation.

It is also more difficult to install or remove Primary, Reference, or Single Reference time servers on the network. You must manually change the approved server list maintained on each server.

To customize your time servers, you can use the SERVMAN utility or SET parameters to set the following parameters:

- ◆ Time Sources
 - Lists the specific time source servers that a server should contact.
- ◆ Configured Sources
 - Specifies that a server should not listen for SAP information from other time source servers.

- ◆ Service Advertising

Disables time source SAP information from being broadcast to the network.

- ◆ Directory Tree Mode

Indicates the server should ignore time sources advertising via SAP if the advertising does not originate on the server's Directory tree. This parameter has no effect if the Configured Sources parameter is turned on.

For detailed information on setting these parameters with SERVMAN or the SET utility, see “Monitoring and Maintaining Time Synchronization” in *Supervising the Network* .

Using a Combination of Communication Formats

You can use both the SAP and custom configuration methods on the same network. However, the custom configuration list that is stored on a server always takes precedence over the SAP information received by the server.

If a server does not have a custom configuration list, SAP information is used for time synchronization.

Hint: On a network where servers are rarely added or reconfigured after initial installation and where the network uses a Single Reference time server, consider using SAP (this is the installation default).

On a network where servers are added or removed frequently, use a custom configuration list.

Summary

Time synchronization is a means to maintain a proper sequence of NDS events when the Directory database is partitioned and replicated.

Time synchronization can be provided by external time sources or internal time sources.

Evaluation

Once you have completed planning time synchronization, review the following questions to evaluate the efficiency of your plan:

- ◆ If your network meets the following conditions, are you using the default settings?
 - ◆ No WAN links exist
 - ◆ Fewer than thirty servers exist
 - ◆ Single time zone exists
- ◆ If consistent time with UTC is important, are you connected to some type of external time source?
- ◆ If you maintain slow WAN links, are you using a configured list?
- ◆ If you have more than seven Primary time servers in the time provider group, are you implementing additional time provider groups synchronized to the same external time source?

Defaults

The defaults for time synchronization are as follows:

Time Server Type	Default
Single Reference	The first NetWare 4 server in the network is set up as a Single Reference time server.
Secondary	All additional servers in the network after the first server are set up as Secondary time servers.

Where to Go from Here

To	Go to
Create an accessibility plan for determining how network resources are accessed and used	Chapter 6, "Creating an Accessibility Plan," on page 107
Create a migration strategy for servers and workstations from a previous version of NetWare or other network operating system	Chapter 9, "Developing a Migration Strategy," on page 173
Create an implementation schedule	"Creating an Implementation Schedule"

Creating an Accessibility Plan

This chapter describes the process used for creating an accessibility plan for your network. The following topics are discussed:

- ◆ “Understanding How Network Resources Are Accessed” on page 109
- ◆ “Determining Access Needs” on page 115
- ◆ “Determining an Efficient Access Control Method” on page 123

The depth of your Directory tree and number of containers has the greatest impact on how network resource are accessed.

Single server environments with only one or two Directory tree layers and limited security concerns might not need to create an accessibility plan.

Introduction

Access to network resources and file system data in a NetWare[®] 4 environment is controlled by the Novell[®] Directory Services[™] (NDS[™]) technology and NetWare operating system. Network resources are all contained in a single-information system represented by the Directory tree.

Each logical and physical resource within the tree is represented as an object that can be accessed and managed by its location within the tree structure.

Network file system data is linked to the Directory tree through Volume objects, and is represented in the tree structure by its relationship to the Volume object.

The Directory tree structure allows resources to be organized in a hierarchical fashion. This hierarchical structure supports intuitive access to, and administration of, network resources and services. In addition, the tree structure allows for easy security administration on a container-by-container basis or single object basis, depending upon your particular needs.

Creating an accessibility plan involves understanding network access in NetWare 4, identifying the access needs for your organization, determining the most efficient configuration, and developing a system for controlling access.

When creating an accessibility plan for your network, it is important to remember that all rights and access control flow down the tree structure. This means that when creating the most efficient accessibility plan for your network, you should consider the Directory tree structure and global access to Directory tree resources.

You should also create an accessibility plan that provides the most effective use of login scripts and global objects for quick access and security throughout the network without high management overhead.

Objectives

You should identify access and security needs for users, applications, and network resources. Afterwards, create accessibility guidelines for designing and configuring login scripts and placing access and administration objects in the tree.

This will enable you to use resource maps, location maps, LAN and WAN topology maps, organizational charts, and guidelines to determine the level of access and security to meet your network's needs.

Prerequisites

- You should have copies of the following planning documents:
 - ◆ Directory tree structure design draft
 - ◆ Resource maps
 - ◆ Location maps

- ◆ LAN and WAN topology maps
 - ◆ Organizational charts
- You should have completed your Directory tree design

Understanding How Network Resources Are Accessed

Because network resources exist in a hierarchical tree structure, accessing a particular object requires information about the object's name and location in the tree. Users and network resources use an object's name to locate and interact with other objects.

Each leaf object has a name to identify it. This name is referred to as the leaf object's *common name (CN)*. For User objects, the common name is the User's login name. Other leaf objects also have common names such as a Printer object name, NetWare Server object name, or Volume object name.

Container objects don't have common names. They are referred to by their Organizational Unit (OU) object name, Organization (O) object name, or Country (C) object name.

Identifying Objects by Name

An object's location in the tree is referred to as its *context*. An object's tree name (Directory name) is identified by the full path from the object's context in the tree to the [Root] of the Directory tree.

Complete Name

The full path of an object's location in the tree, which is from its current location up to the [Root] object, forms the object's *complete name*, or *distinguished name (DN)*.

Note: The term distinguished name is commonly used interchangeably with complete name.

For example, a complete name or fully distinguished name (DN) for the User object ESAYERS could be

.CN=ESAYERS.OU=SALES.OU=HQ.O=ACME

Note: The objects in the name are separated by periods, similar to the backslash (/) used in DOS paths. A leading period is used. A leading period directs NDS to ignore the current context of the object and resolve the name at the [Root] object. A trailing period cannot be used.

Partial Name

An object's current location in the Directory tree is called the *current context* or *name context*. The Directory name of an object's current context relative to other Directory objects is referred to as its *partial name* or *relative distinguished name (RDN)*.

Note: The term relative distinguished name is commonly used interchangeably with partial name.

The partial name is a subset of an object's complete name. It allows resources to search for and locate other Directory objects by their relative context (tree location) to each other. This makes referencing objects near the requesting object simpler and easier.

When using an object's partial name only the portion of that object's complete name that is *not* common between another objects is used.

For example, a partial name for the User object ESAYERS relative to other objects in OU=SALES would be

CN=ESAYERS .

The partial name of the User object ESAYERS that has a complete name of

CN=ESAYERS . OU=SALES . OU=HQ . O=ACME

relative to a Printer object with the complete name of

CN=PDLJ4_02 . OU=PROD . OU=MFG . O=ACME

is **CN=ESAYERS . OU=SALES . OU=HQ .**

Note: The objects in the name are separated by periods, similar to the backslash (/) used in DOS paths. A leading period and trailing period can be used. A leading period directs NDS to ignore the current context of the object and resolve the name at the [Root] object. A trailing period allows a network resource to select a new context when resolving an object's complete name at the [Root].

A partial name must still be resolved at the [Root] object. This is done by adding a trailing period at the end of the partial name. Adding a trailing period forces NDS to identify the object's context and automatically resolve the rest of the object's complete name.

Typeful and Typeless Naming

An object's distinguished name consists of different object types, such as common name (CN), Organizational Unit (OU) objects, and Organization (O) objects. When the abbreviations of these object types are used in an object name, it is referred to as an object's *typeful name*. For example,

CN=ESAYERS . OU=SALES . OU=HQ . O=ACME

In many cases, the abbreviated object types can be omitted when referencing a Directory object. This kind of naming is referred to as an object's *typeless name*. For example,

ESAYERS . SALES . HQ . ACME

If the object types are not provided in an object's complete name, NDS identifies the attribute type for each object in the name.

Name Length and Tree Depth

Maintaining an appropriate depth for your environment enables easier access to and management of the network.

A Directory tree should maintain a depth of four to eight levels. As the complexity of your environment increases, either in numbers of objects or in the number of locations under single management, then the tree depth can easily increase to accommodate those conditions.

Nevertheless, a limitation of the DOS command line utilities impose a maximum context length of 255 characters. If shorter Organizational Unit (OU) names are used a deeper tree can be designed. However, the greater the depth of the tree, the more complex access to network resources may be required.

The total character count is established by using an object's complete name in its typeful name format. This includes the object type abbreviation, equal sign (=), and periods (.).

When leaf objects are created, they are checked to ensure that the maximum name length of the complete name isn't exceeded.

However, it is possible to rename a leaf object and cause the complete name to exceed 255 characters. The following example shows an acceptable complete name:

```
CN=JSMITH.OU=SALES.OU=HQ.O=ACME.ACMECORP
```

Identifying Directory Objects by Location

Network resources search and navigate the Directory tree to locate objects in their particular context. For example, a user can navigate from one container to another by *changing context*s. This doesn't mean that the User object for that user is moved to a different context in the tree, but that the user's view of the Directory tree is changed to a different context.

Nevertheless, once a user changes context, the names of objects in the tree for that user's User object are now relative to the user's current context. This allows users to navigate the tree to find and access Directory objects in their particular containers.

NetWare 4 provides both text-based and graphical utilities for navigating the tree.

A network resource can also use the complete name or partial name of an object for searching the Directory tree.

For the User object ESAYERS to access a Volume object located in the HQ container, the following map statement should be used:

```
MAP drive_letter :=CN=servername _volumename .OU=HQ.:
```

For example, to map the APPS volume of the server SALES1 to drive letter G:, type,

```
MAP G:=CN=SALES1_APPS.OU=HQ.:
```

Note: The typeful or typeless can be used for accessing other objects in the tree.

Using Access-Related Objects

Access-related objects help simplify tree navigation and access to commonly-used network resources.

Alias Object

An Alias object is a pointer to an actual resource object in the tree. An Alias object can point to either a container object or a leaf object.

For example, users in the SALES Organizational Unit object can access a Printer object located in the HQ Organizational Unit object through an Alias object in their same container. This allows users to reference the actual printer by using only the common name of the Alias object.

Alias objects can also point one Organizational Unit object to another Organizational Unit object. This allows the access rights to objects within the aliased container to apply to users in the container holding the Alias object.

For example, you might create an Organizational Unit object that holds a group of application servers. Users outside of this Organizational Unit object might also need access rights to the application servers. If you create an Alias object in the users' container to the container holding the application server, the users have the same access rights to the application servers that exist for the container that holds the servers.

Naming Alias Objects

You might want to give an Alias object a name that indicates that it is a pointer to a primary object. For example, the name might include the word *Alias* such as ALIAS_MKT_SRV1.

Inversely, you might not want to distinguish the Alias object from the primary object. Users may not need to know the difference, and adding *Alias* to the name might only confuse them.

Relationship to Primary Objects

It is important to understand how Alias objects relate to the primary objects they point to. Alias objects exist in two different states: *dereferenced* and *non-dereferenced* (or *referenced*).

When an Alias object is dereferenced, operations performed on the Alias object point to the properties of the primary object. This means that when changes are made to the Alias object, the changes are actually made to the primary object.

When an Alias object is non-dereferenced, operations performed on the Alias object affect only the Alias object itself. Actions such as moving, renaming, or deleting an Alias object are automatically non-dereferenced.

If you delete the primary object of an Alias object, the Alias object is automatically deleted.

Directory Map Object

A Directory Map object allows objects in one container to access file system directories or Volume objects located in another container. This is useful when a particular application or file can only exist on a single volume, but is accessed by objects in many containers.

For example, users in the SALES Organizational Unit object can access a Directory Map object pointing to a database application stored on a volume located in the HQ Organizational Unit object. This allows users to reference the actual database volume by using only the common name of the Volume object.

Note: Directory Map objects can point to a specific Volume object or file system directory on the volume.

The Directory Map object can manage mappings in container or user login scripts. For example, if many different container or user login scripts maintain individual drive mappings to a particular application directory, they all have to be changed individually when the application directory is changed or the application is updated to a new directory. Conversely, if container or user login scripts referenced a single Directory Map object for the application directory, any changes would be reflected only in the Directory Map object itself.

When you assign the Directory Map object a path to the files or applications you are referencing, you must also grant each User object Read and File Scan rights to the files or applications in the directory. You can do this by granting Read and File Scan rights to the Directory Map object, and then make each user security equivalent to the Directory Map object. You might also assign file rights to each Organizational

Unit object. Users objects are automatically granted security equivalency to their particular Organizational Unit objects.

Global Group Objects

A Group object contains users from any container in the Directory tree. It can be located in any container and be granted any rights. This allows you to create a global Group object to regulate global access to the Directory tree for a specific set of users.

For example, you can create a managers group or publications group that requires the same access to network resources and include all the necessary users in the Group object. This type of Group object allows for a single point of access management to a single network resource or container of resources.

Group objects allow you to grant users in Organizational Unit objects specialized rights assignments. Therefore, you can manage a much smaller subset of users within the Directory tree.

Determining Access Needs

When determining the particular network access needs for your organization, consider the following issues:

1. What types of network connections are needed?
2. What type of Novell Client™ Software is being used?
3. Are users static or mobile?
4. What network resources do users access and how are they shared?
5. Is bindery services needed?

Identifying Network Connection Types

NetWare 4 supports three types of network connections:

- ◆ Attached (not logged in)

Once the Novell Client Software is loaded, NetWare 4 allows users and other network resources to browse the Directory tree and locate objects. Limited information is available about each object; however, no operations can be performed on the objects. No licensed connection is used.

- ◆ Authenticated

When a request is made to a Directory object, an authenticated connection needs to be established. This is referred to as a *pass-through operation*. Logging in to the network or changing an object property is an example of a pass-through operation that requires authentication before it is allowed. No licensed connection is used. See “Authentication” on page 123 for more information.

- ◆ Licensed

Once an authenticated connection is established, operations such as mapping a network drive or capturing a printer port can be performed. When a request for access to a network resource is initiated, such as mapping a network drive, a licensed connection is used.

Note: Administrators can browse and manage the Directory tree and not use a licensed connection by loading utilities from their workstation's local drives.

Identifying Novell Client Types

NetWare 4 supports for the following Novell Client types:

Client Operating System	Explanation
DOS and Windows	<p>NetWare 4 provides full NDS support for DOS and Windows clients running Novell Client Software. The Novell Client Software supports container and user login scripts in DOS and Windows.</p> <p>NetWare 4 also provides full NDS support for DOS and Windows clients running the NetWare DOS Requester software. The NetWare DOS Requester™ supports container and user login scripts within DOS and Windows.</p>
NFS/UNIX	<p>NetWare 4 provides full bindery-based login for NFS* clients. All resources are accessed through bindery services. NFS clients do not support container or user login scripts. All drive mappings and port captures are maintained through individual user profiles within the operating system.</p>

Identifying User Types

All networks support one or both types of network users:

- ◆ Local

Local users are static, meaning that they commonly access network resources from the same Directory context. Local users require that objects they commonly access are placed in close proximity to their User object in the tree. In addition, physical resources such as printers and applications are connected or stored on the server that users attach to.

- ◆ Mobile

Mobile users commonly access network resources from varying parts of the network or Directory tree. They may be physically located at a different site or logically located in a different part of the tree. Easy access to physical network resources and Directory tree information should be considered.

Mobile users require consistent and common network configuration. Directory objects should be placed in a common fashion throughout the Directory tree. Use of access objects such as Alias objects, Directory Map objects, or Group objects should be consistent across the Directory tree.

Application and workgroup servers throughout the network should maintain identical file structures if possible.

Placing an Alias object close to the [Root] for mobile users makes login and authentication easier. This eliminates the need for users to remember their complete name (distinguished name).

Efficient placement of partitions in the network helps mobile users find Directory information.

Identifying Global and Shared Resources

Global and shared resources are common in network environments. These resources are used by users across LAN and WAN links. Examples of global resources are customer databases, applications, e-mail, calendars, telephone books, printers, and application servers. Considerations should be made to ensure efficient and intuitive access to these resources.

You might need to create special containers close to the [Root] that contain Alias objects or Directory Map objects of global resources. For example, you can create a container that has Directory Map objects to a common application suite.

You could also create a container with an Alias object of each network user so that global applications such as e-mail would reference a single location for Directory information. This container could be partitioned and replicated across the network. Because Alias objects are extremely small objects with few updates to them, synchronization is very efficient.

When a user authenticates to the network, the profiles and scripts for that user are executed. If any of those are not kept locally, the login process retrieves those profiles and scripts across LAN or WAN links.

Keeping copies of profiles and scripts close to the user decreases the time needed to complete the authentication process.

Leaf objects such as Volume objects and Printer objects are easiest to access when they are in the lowest container level that incorporates all the objects that need access to them.

For example, if a printer is used by two different departments (which have separate Organizational Unit containers), place the Printer object at a level above the two containers for those departments.

Identifying Bindery Services Needs

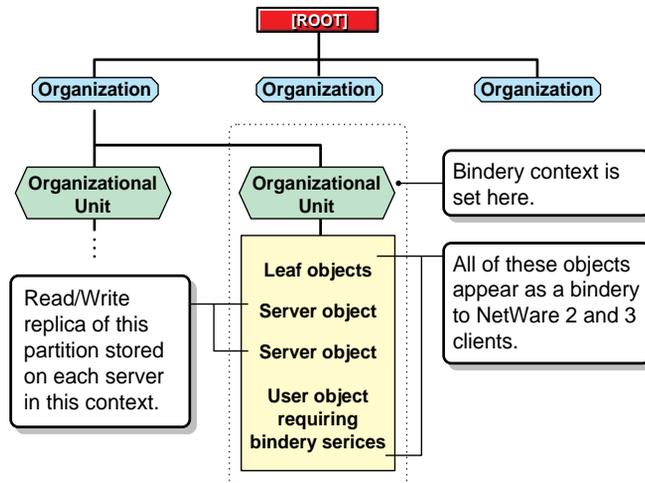
Some applications and services which run in the NetWare 4 environment do not currently take full advantage of the NDS technology. To enable users to access these services from the NetWare 4 environment, Novell created *bindery services*.

With bindery services, NDS imitates a flat structure for leaf objects in an Organization or Organizational Unit object. When bindery services is enabled, all objects in the specified container can be accessed by NDS objects and by bindery-based servers and client workstations.

Important: Bindery services applies only to leaf objects in a specified container object.

The following figure illustrates bindery services when an Organizational Unit object is specified as the bindery context.

Figure 6-1
Bindery Services in a Directory Tree



A writable replica of the partition that includes the container object to be set as the bindery context must be stored on each server you want bindery services enabled on. However, by default, only the first three servers installed on a partition receive a replica of the partition during the installation process and subsequently support bindery services.

You can add replicas to other servers if needed for bindery services. If a read/write or master replica is not present, use the NDS Manager option in NetWare Administrator or PARTMGR to add one to the server.

Note: If a bindery context is not set, NDS cannot support bindery services.

Bindery services is server-centric. If a client workstation does a bindery login, the login script comes from the server that the client is logging in to. Any changes to the user's bindery login script are made on a single server and are not distributed to other servers.

You cannot disable bindery services if someone is logged in through bindery services, and bindery objects are always available unless bindery services is disabled.

Bindery services allows NetWare 4 servers to support the following bindery-based resources:

- ◆ Bindery objects class
- ◆ Bindery-based Novell Client Software
- ◆ Users
- ◆ Groups
- ◆ Queues
- ◆ Print servers
- ◆ Profiles
- ◆ Bindery programs

You should identify what applications and network resources are bindery-based, such as jetdirect printers or client workstations.

Each NetWare 4 server supports up to sixteen different bindery context settings. If bindery-based applications and resources exist in your network, you should consider basing your accessibility guidelines on the bindery context for each resource.

Determining What Objects to Create

If you or a service require the bindery user GUEST, you must create GUEST in the Directory database.

During installation, a bindery object SUPERVISOR is created but is not used with NDS. The NDS utilities do not display this object. This object is intended to be used with bindery services and to enable access to the server through a bindery login. Once bindery services is enabled, you can use this object to log in to the server, providing you log in as a bindery object.

You can create an NDS User object SUPERVISOR and assign ADMIN-equivalent rights to it in NDS. However, the bindery object and the NDS object are unique and separate objects even though they are identified by the same name.

After installing the NetWare server, you can use a migration utility to convert bindery user accounts to NDS User and Group objects. If you do, all users except SUPERVISOR and all groups are updated to NDS objects. The user SUPERVISOR is migrated, but with supervisory rights for that server's file system and bindery context only. SUPERVISOR does not appear as an Directory object.

Identifying Possible Limitations

While bindery services will allow users to access bindery-based applications and resources, you need to be aware of its limitations.

Information Limitations

Some Novell Directory Services information is not available to users through bindery services. This information includes, but is not limited to, the following:

- ◆ E-mail name
- ◆ Phone number

- ◆ Print job configurations
- ◆ Aliases
- ◆ Profiles
- ◆ NDS login scripts

Partitioning Limitations

The bindery context for a server can be set to a container that is part of a partition stored on a different server. But, before you can use bindery services, you must place a writable replica of the partition that includes the bindery context on the bindery services-enabled server.

If you set the bindery context for a server to a container object that is not part of a writable replica on that server, users will not be able to log in through bindery services.

Changing Context Limitations

Avoid changing a server's bindery context once you set it. Changing a server's bindery context leaves users in the original context without access to bindery services. Changing the server's bindery context can also cut off access to print queues.

Network Traffic Concerns

Needing bindery services on all or nearly all servers to support bindery-only aware applications places a extra load on the network due to the replication traffic that is exchanged between all instances of a partition.

To reduce network traffic, you could:

- ◆ Partition farther down the tree so that fewer servers hold the replica of any partition.
- ◆ Move the bindery-only aware applications to certain servers and then place replicas only on those servers.
- ◆ Upgrade old applications, or purchase new Directory-aware applications.

Determining an Efficient Access Control Method

Access control is an integral part of Novell Directory Services and the NetWare file system structure. It determines what actions users can perform and what information and resources are available.

An efficient security structure is easily implemented because most of the necessary rights are automatically assigned as Directory objects are created.

Administrators can control users and groups needing access to resources, such as data and programs that reside in files and directories. They can also protect all the objects at the server level from unauthorized access.

Access control to Directory objects is maintained through the following set of features:

- ◆ Authentication
- ◆ NDS and file system security
- ◆ Login and profile scripting
- ◆ Administrative objects

Authentication

When a Novell Client makes a request for access to a service on the network, such as logging in, a server begins a process called *authentication*. Authentication validates a client request by attaching a unique code to each request. This unique code is then used to identify the following information about each request:

- ◆ The source of the request
- ◆ What session the request pertains to
- ◆ If any information was counterfeited from another session
- ◆ If the request data is corrupt or has been tampered with

For example, authentication occurs when a network user makes a login request. NDS returns a unique code to the user request that is attached to the user's login information (password, workstation address, and time). Based on the current context and the login name, authentication identifies the User object to other servers in the tree and verifies that the object has rights to use certain resources.

Authentication enables a NetWare 4 network to support a single login for the entire network of servers.

Authentication allows a user who has logged in to the network to access any resource in the network that the user has rights to. If a user lacks sufficient rights, access is denied. Authentication checks a user's rights to both Directory and file system resources.

NDS and File System Security

NetWare 4 supports two divisions of security for the Directory tree and file system.

Novell Directory Services security affects management of the Directory tree and its objects. This security is used for managing the Directory objects and their properties, such as access between Directory objects and their properties, access to login scripts, etc.

The NetWare file system security affects how Directory objects can access files and directories on network volumes. This type of security provides control of the application programs and data files on network servers.

NetWare 4 file system security is essentially the same as file system security used in previous versions of NetWare. Some new attributes have been added for features such as data compression and data migration.

NDS security and file system security are based on the same principles but function separately. This allows for single or divided administration of network resources and network data.

The common principles for NDS and file system security are

- ◆ Trustee assignments
- ◆ Inheritance
- ◆ Inherited Rights Filter (IRF)
- ◆ Security equivalence
- ◆ Effective rights

Trustee Assignments

A trustee assignment determines the access Directory objects have to other Directory objects and their properties, and access to file system directories and files. These assignments are made by explicitly assigning rights to a Directory object and its properties, and file system files and directories.

Some trustee assignments are automatically made at installation, and when certain Directory objects such as User objects and NetWare Server objects are created. See “Default Trustee Assignments” on page 144 for more information.

Trustee assignments have the following characteristics:

- ◆ Trustee assignments flow from the [Root] to lower branches in the Directory tree or from the root to the lower file directories in the file system.
- ◆ An explicit assignment at a lower level replaces all trustee assignments made at higher levels in the Directory tree or file directory.
- ◆ Selected property rights override rights assigned through the [all property rights] attribute.
- ◆ Every Directory object, file directory, and file maintains a trustee list of all User objects, Group objects, or Organizational Role objects that have access rights to it.
- ◆ Trustee assignments for the file system are stored in the directory entry table (DET), while trustee assignments for Directory objects and properties are stored in the ACL (Access Control List).

Inheritance

Because the Directory tree and file system are hierarchical tree structures, rights assigned in the Directory tree or file system flow down through the tree. This is referred to as *inheritance*. Inheritance allows rights assigned at upper levels in the tree or file system to flow down to subordinate levels. Rights received from upper levels are referred to as *inherited rights*. These inherited rights flow down without specific trustee assignments.

Inherited Rights Filter

Inherited rights are controlled by blocking specific rights with an *Inherited Rights Filter* (IRF).

In the Directory tree, objects automatically receive, or inherit, rights granted to their parent objects. The IRF is used to block any or all of the inherited rights from flowing down to subordinate objects.

It is important to remember however, that an IRF cannot grant rights; it can only block rights assigned to objects at higher levels in the tree. Nevertheless, an IRF can be enabled for every file, directory, Directory object, and object property.

The Supervisor object right and property right can be blocked by an IRF. However, the Supervisor rights to files and directories can't be blocked by an IRF.

Security Equivalence

Security equivalence allows you to assign rights through association. This means that an object can acquire rights by its assigned relationship to another object, such as, containers, groups, organizational roles.

Security equivalence allows an object to be equivalent in rights to another object. Every object is security equivalent to the [Root] object and the [Public] object trustee by default. This ensures that all objects can search and navigate the Directory tree.

Security equivalent rights flow down through the Directory tree independent of any other trustee assignments. Therefore, rights assigned to an object do not affect rights received through security equivalence. For example, a User object could be made security

equivalent to a Group object with Supervisor rights to all objects in the tree. Any explicit rights assigned to the User object in a particular Organizational Unit object will not affect the rights received through security equivalence.

Implied Security Equivalence

Every object is security equivalent to all container objects that are a part of its complete name (distinguished name). This is known as *implied security equivalence*.

Implied security equivalence is a characteristic of the Directory schema and cannot be modified. Because of this, the implied security of an object is not viewable by NetWare utilities.

Security equivalence is not transitive. For example, a User object that is security equivalent to User object ADMIN receives security equivalences that the User object ADMIN might have with other objects.

Security Equivalence and Inheritance

Security equivalence differs from inheritance in that inheritance allows rights to flow down the tree from parent to child objects until rights are blocked by an IRF. Security equivalence, however, applies only to rights granted explicitly to the objects that one maintains security equivalence to.

A simple rule to remember is that inheritance can be blocked by enabling an IRF, but security equivalence or trustee assignments cannot be blocked. Security equivalence and trustee assignments must be explicitly granted and revoked.

This is important to understand because all objects in an Organizational Unit object container are automatically security equivalent to the Organizational Unit object. Enabling an IRF for the Organizational Unit object does not affect the rights received from the Organizational Unit object through security equivalence.

Effective Rights

The actual rights that an object has is dependant on a combination of explicit trustee assignments, inheritance, and the IRF. This combination determines an object's *effective rights*.

An object's effective rights are what control its access to another object and that object's properties. These rights define what the object can actually do at a particular level in the Directory tree or file system.

A User object might have explicit trustee assignments at the Organizational level, but may have very different rights at lower Organizational Unit object levels because of an IRF. This is important to understand when calculating the effective rights for a given object.

NDS Security

Once a user has logged into the network, access to leaf and container objects is determined by the NDS security structure. At the foundation of NDS security is the Access Control List (ACL).

The ACL is a property of every Directory object. It defines who can access the object (trustees) and what each trustee can do (rights).

Each object listed in an ACL can have different rights to that object's properties. For example, if ten users are listed in a Printer object's ACL as trustees, each of those ten users can have different rights to that Printer object and to its properties. One object might have the Read right, another might have the Delete right, etc.

To change the trustee's access to an object, you would change the trustee's entry in the object's ACL. Only trustees with the Write right for the ACL property can change the trustee assignments or the Inherited Rights Filter.

The ACL is divided into two types of rights:

- ◆ Object rights
 - Defines an object's trustees and controls what the trustees can do with the object.
- ◆ Property rights
 - Limits the trustee's access to only specific properties of the object.

In summary, object rights define who can access the object and what they can do with the object. Property rights further refine the level of access by specifying the object properties that can be accessed.

Object Rights

Object rights control what trustees of an object can do with that object. Object rights control the object as a single entity in the Directory tree, but do not allow the trustee to access information stored in that object's properties (unless the trustee has the Supervisor object right, which also includes the Supervisor property right).

The following table describes object rights you can assign to a trustee.

Right	Description
Browse	Grants the right to see the object in the Directory tree. Also allows a user performing a search to see the object if it matches the search value. (This is true only when comparing the base object class or partial name; otherwise, the Compare right for property objects is required.)
Create	Grants the right to create a new object within a container object in the Directory tree. This right applies only to container objects because leaf objects cannot contain other objects.
Delete	Grants the right to delete an object from the Directory tree. However, a container object cannot be deleted unless all the objects in the container are deleted first. The Write right for all existing object properties is also needed to delete objects.
Rename	Grants the right to change the partial name of the object, in effect changing the naming property. This changes the object's complete name.
Supervisor	Grants all rights to the object and to all its properties. Anyone with Supervisor rights to an object has access to all of its properties. The Supervisor right can be blocked with the Inherited Rights Filter (IRF).

Property Rights

While object rights allow you to see an object, delete an object, create a new object, etc., only the Supervisor property right allows you to see the information stored in an object's properties.

To see the information in an object's properties, you must have the correct property rights. Property rights control access to each property in an object.

Property rights apply only to Directory object properties, not to the objects themselves. NDS allows you flexibility in deciding what property information others can access.

The following table describes property rights you can assign to a trustee.

Right	Description
Add or Delete Self	<p>Allows you to add or remove yourself as a value of the property, but you cannot change any other values of the property.</p> <p>This right is only used for properties where your User object can be listed as a value, such as group membership lists or mailing lists.</p> <p>This right is included in the Write right; that is, if the Write right is given, Add or Delete Self operations are also allowed.</p>
Compare	<p>Allows you to compare any value to an existing value of the property. The comparison can return True or False, but cannot give the value of the property.</p>
Read	<p>Allows you to read the values of the property.</p> <p>This right includes the Compare right; that is, if the Read right is given, Compare operations are also allowed.</p>
Supervisor	<p>Gives you all rights to the property. This Supervisor right can be blocked by an object's Inherited Rights Filter (IRF).</p>

Right	Description
Write	<p>Allows you to add, change, or remove any values of the property.</p> <p>This right includes the Add or Delete Self right; that is, if the Write right is given, Add or Delete Self operations are also allowed.</p> <p>The Write right to the Access Control List (ACL) property is the same as giving the Supervisor right to the object—the right to grant rights.</p>

Property rights can be assigned in one of two ways,

- ◆ All Properties option

Assigns the rights you choose to all properties for the object. For example, the Read All Properties right setting would allow you to view the value of all properties for an object.
- ◆ Selected Properties option

Assigns rights only to the properties that you have specified. Granting rights to specific properties overrides any rights granted through the All Properties option. This allows you to define general rights assignments for a group of objects and specific property settings for a select object.

NetWare File System Security

NetWare file system security exists at the server level. The server stores volumes that contain the directories which contain files. The file system security does not transition into the NDS security structure.

Nevertheless, access to the NetWare file system is controlled by the same principles as NDS security. The basic principles include trustee assignments, inheritance, and security equivalence. The file system also uses Inherited Rights Filters (IRF) which participates in determining the effective rights.

Note: Previous versions of NetWare referred to the file system IRF as an Inherited Rights Masks (IRM).

There are however a few minor difference between NDS and file system security:

- ◆ NDS has ten access rights divided into two groups:
 - ◆ Objects
 - ◆ Properties
- ◆ Rights do not flow from NDS into the file system except in the case of the Supervisor [S] object rights to the NetWare Server object. This grants the trustee Supervisor [S] file system rights to the root of all server volumes.
- ◆ The Supervisor [S] object rights can be blocked by the IRF. The Supervisor [S] file system rights cannot be blocked by the IRF.

File System Access Rights

Once a user has logged into the network, access to files and directories is determined by the NetWare file system security structure. At the foundation of NetWare file system security is the directory entry table (DET).

The DET stores access information about directories and files. It contains information about a volume's file and directory names, and properties.

For example, an entry might contain the following:

- ◆ Filename
- ◆ File owner
- ◆ Date and time of last update
- ◆ Trustee assignments

Before you can access files and directories, you must have sufficient file system access rights.

The following table lists the available file system rights for making trustee assignments in NetWare 4:

Right	Allows You To
Access Control	Add and remove trustees and change rights to files and directories.
Create	Create subdirectories and files.
Erase	Delete directories and files.
File Scan	View file and directory names in the file system structure.
Modify	Rename directories and files, and to change file attributes.
Read	Open and read files, and to open, read, and execute applications.
Supervisor	Grant all rights listed in this table.
Write	Open, write to, and modify a file.

There are three rights that you should use with caution.

◆ **Supervisor [S]**

The Supervisor [S] right grants all privileges to files and directories and it cannot be filtered.

Users with Supervisor [S] rights can make trustee assignments and grant all rights to other users.

◆ **Access Control [A]**

The Access Control [A] right allows users to make trustee assignments but they can only grant the same rights that they possess. Access Control [A] also allows users to modify the IRF.

◆ **Modify [M]**

The Modify [M] right allows users to change files and directories. It also allows users to change file system attributes.

File System Attributes

File system attributes assign rights to individual directories or files. Some attributes are meaningful only when applied at the file level, but some apply to both the directory and file levels.

Be careful when assigning directory and file attributes. The attribute applies to *all* users.

For example, if you assign the Delete Inhibit attribute to a file, no one, including the owner of the file or the system supervisor, can delete the file. But any trustee with the Modify right can change the attribute to allow deletion.

The following table lists and explains the rights stored in the directory entry table (DET) for files and directories:

Table 6-1

Directory and File Attributes

Attribute code	Description	Applies to
A	Archive Needed identifies files that have been modified since the last backup. This attribute is assigned automatically.	Files only
Cc	Can't Compress indicates the file cannot be compressed because of limited space savings.	Files only
Co	Compress indicates that a file is compressed.	Files only
Dc	Don't Compress keeps data from being compressed for all files in a directory or individual files. This attribute overrides settings for automatic compression of files not accessed within a specified number of days.	Directories and files
Di	Delete Inhibit means that the file or directory cannot be deleted. This attribute overrides the Erase trustee right.	Directories and files
Dm	Don't Migrate prevents files and directories from being migrated from the server's hard disk to another storage medium.	Directories and files
Ds	Don't Suballocate prevents data from being suballocated.	Files only
H	The Hidden attribute hides files and directories so they can't be listed using the DIR command. A user with File Scan rights can use FILER or the NDIR command to list directories and files with the Hidden attribute.	Directories and files
I	Index allows large files to be accessed quickly by indexing files with more than 64 File Allocation Table (FAT) entries. This attribute is set automatically.	Files only
Ic	Immediate Compress sets data to be compressed as soon as a file is closed. If applied to a directory, every file in the directory is compressed as each file is closed.	Directories and files
M	Migrate indicates that a file has been migrated from the server's hard disk to another storage medium.	Files only
N	Normal indicates the Read/Write attribute is assigned and the Shareable attribute is not. This is the default attribute assignment for all new files.	Directories and files
P	Purge flags a file or directory to be erased from the system as soon as it is deleted. Purged files and directories cannot be recovered.	Directories and files

Attribute code	Description	Applies to
Ri	Rename Inhibit prevents the file or directory name from being modified.	Directories and files
Ro	Read Only prevents a file from being modified. This attribute automatically sets Delete Inhibit and Rename Inhibit.	Files only
Rw	Read/Write allows you to write to a file. All files are created with this attribute.	Files only
Sh	Shareable allows more than one user to access the file at the same time. This attribute is usually used with Read Only.	Files only
Sy	The System attribute hides the file or directory so it can't be seen by using the DIR command. It can be seen if a user with File Scan rights uses FILER or the NDIR command. System is normally used with operating system files, such as DOS system files.	Directories and files
T	Transactional allows a file to be tracked and protected by the Transaction Tracking System (TTS).	Files only
X	The Execute Only attribute prevents the file from being copied, modified, or backed up. It does allow renaming. The only way to remove this attribute is to delete the file. Use the attribute for program files such as .EXE or .COM. Make a copy of a file before you flag it as Execute Only, so you can replace the file if it becomes corrupted.	Files only

Login and Profile Scripting

Login scripts define the user's drive mapping, capture statements, and variable settings. They also invoke menus and applications. For ease of network administration, users and the resources they use should be placed together within Organizational Unit objects.

Four Types of Login Scripts

When a user logs in, the LOGIN utility executes the appropriate login scripts. Four types of login scripts are available, and they can be used separately or together to create a custom environment for your users. All but the default login script are optional.

Login scripts are executed in the given order:

◆ **Container login script**

This sets the general environments for all users in a container. The LOGIN utility executes container login scripts first. A user can use only one container login script.

Note: A container login script replaces the system login script from NetWare 3™.

◆ **Profile login script**

This sets environments for several users at the same time. The LOGIN utility executes a profile login script after the container login script.

A user can be assigned only one profile login script, but can specify other profile login scripts on the command line. Several users can use the same profile login script.

◆ **User login script**

This sets environments specific to a single user, such as printing options or a username for e-mail. The LOGIN utility executes the user login script after any container and profile login scripts have executed.

A user can have only one user login script.

◆ **Default login script**

This is precoded into the LOGIN.EXE command and is not editable. It executes if a user doesn't have his or her own user login script, even if a container or profile login script exists.

The default login script is executed for all users (including user ADMIN) unless you create a user login script. The default login script contains only essential commands such as drive mappings to the NetWare utilities.

If you don't want to create any user login scripts and you don't want the default login script to execute for any users, you can disable the default login script by including the NO_DEFAULT command in the container or profile login script.

To use the login script from an Organization, Organizational Unit, or Profile object, users must have the Browse right to the object and the Read right to the object's Login Script property.

For more information on Browse or Read rights for a file, object, or property, see “Browsing” and “Rights” in *Concepts*.

Planning Effective Login and Profile Scripts

Maintaining many user login scripts can be time consuming. Try to include as much customizing information as possible in the container and profile login scripts, which are fewer in number and easier to maintain.

For example, if all users need access to the NetWare utilities in the same volume, put the search drive mapping to that volume in a single container login script rather than in every user login script.

Create profile login scripts if there are several users with identical login script needs.

Finally, in user login scripts, include only those individual items that can't be included in profile or container login scripts.

Since up to three login scripts can execute whenever a user logs in, conflicts can occur. If this happens, the last login script to execute (usually the user login script) overrides any conflicting commands in a previous login script.

Login scripts are properties of objects. The following table shows which objects can contain which login scripts.

Object	Type of Login Script
Organization	Container
Organizational Unit	Container
Profile	Profile
User	User

The following conventions can help you plan effective login scripts.

Table 6-2

Login script conventions

Subject	Conventions
Minimum login scripts	No minimum. All four types of login scripts are optional. Login scripts can have only one line or they can have many. There are no required commands for login scripts.
Case	Either uppercase or lowercase is accepted. Exception: identifier variables enclosed in quotation marks and preceded by a percent sign (%) must be uppercase.
Characters per line	150 characters per line is maximum; 78 characters per line (common screen width) is recommended for readability.
Punctuation and symbols	Type all symbols (#, %, , _) and punctuation exactly as shown in examples and syntax.
Commands per line	<p>Use only one command per line. Start each command on a new line; press <Enter> to end each command and start a new command.</p> <p>Lines that wrap automatically are considered one command.</p> <p>WRITE command output displays better if WRITE is repeated at the beginning of each wrapped line.</p>
Sequence of commands	<p>Generally, enter commands in the order you want them to execute, with the following restrictions:</p> <ul style="list-style-type: none"> ◆ ATTACH commands must precede related MAP commands to avoid prompting the user for a username/password during login. ◆ If you use # to execute an external program, it must follow any necessary MAP commands. ◆ If sequence is not important, group similar commands, such as MAP and WRITE commands, together to make the login script easier to read.
Blank lines	Blank lines don't affect login script execution. Use them to visually separate groups of commands.
Remarks (REMARK, REM, asterisks, and semicolons)	Lines beginning with REMARK, REM, an asterisk, or a semicolon are comments, which don't display when the login script executes. Use remarks to record the purpose of each command or group of commands.
Identifier variables	Type identifier variables exactly as shown. For the value of an identifier variable to be displayed on the workstation's screen as part of a WRITE command, you must enclose the identifier in quotation marks and precede it by a percent sign (%).

Global Login Scripts

Netware 4 does not use a global system login script. Each Organizational Unit object you create has its own login script (container login script). The order of execution of login scripts is as follows:

1. Container login script, if present
2. Profile login script, if used
3. User login script or the default login script if no other script is available

If you want to create a more global login script and include users from multiple Organizational Unit objects, you could use the Profile object to set up a specific environment for a group of users. A Profile object provides an additional set of drive mappings to what is specified in a container login script.

Creating Location Login Scripts

A Profile object can also be used to determine resource allocation based on location. For example, suppose each floor of your company has three printers and three print queues, and you want to be able to assign a particular group of users to a specific print queue. You can use a Profile object to capture to a particular print queue. The users whose profile attribute has been specified will automatically capture to that print queue.

Creating Special-Function Scripts

You can create a Profile object for a special function script, such as one to assign access for applications. For example, you can create a profile script that will be used by backup administrators only. This script may give these users a specific drive assignment to backup software and utilities.

Administrative Objects

The following objects help administer access to the network:

- ◆ User object ADMIN
- ◆ Organizational Role object

User Object ADMIN

The first time you log in to a new Directory tree, you log in as the User object ADMIN—the only User object created during the NetWare 4 installation process. ADMIN is created when you first set up a Directory tree but not when you later add other servers to an existing tree.

ADMIN is assigned all rights (including the Supervisor right) to every object and property in the Directory tree. This gives ADMIN complete control of the Directory tree.

Hint: When you first log in to a new Directory tree, you may want to create a User object and assign that object Supervisor rights to ensure that you have more than one object with sufficient rights to completely control the tree. Such an object can be critically important if the ADMIN object is deleted accidentally.

When it is created, ADMIN is assigned the Supervisor object right to the NetWare Server object. This gives ADMIN the Supervisor right to the root directory of all NetWare volumes attached to the server, so ADMIN can manage all directories and files on every volume in the Directory tree.

ADMIN does not have any special significance like that of SUPERVISOR in previous versions of NetWare. ADMIN is granted rights to create and manage all objects simply because it is the first object created.

You can rename or delete ADMIN at any time; however, you should assign another User object the Supervisor object right to the [Root] object before you delete ADMIN.

Warning: Never delete ADMIN without having assigned the Supervisor right to another User object. Neglecting to do so can be disastrous because you eliminate supervising control of the Directory tree. Restoring access to the tree can only be accomplished with the assistance of Novell Technical Support.

This warning also applies to other sections of the Directory tree where you have a User object ADMIN defined. At each level of the tree where you have ADMIN defined, be sure you also have a User object with explicit Supervisor rights.

It is also important to remember that rights can be granted at a container, and they can also be taken away. If all rights are filtered at a container and there is not a user in that container with all rights, then that container is without full administrative rights. This can cause problems.

Also note that if you create a User object and assign it security equivalence to User object ADMIN and then delete ADMIN, the new User object loses the security equivalence.

Organizational Role Object

The Organizational Role is similar to a Group object. The basic difference is that a Group object is generally used in a login script and is activity oriented (such as for accessing an application on your server). Organizational Role objects are not used in login scripts and are more suited to creating administrators containing a small number of occupants. The Organizational Role object has an attribute known as role occupant.

An occupant can be moved in and out of the Organizational Role quickly to facilitate short term assignments. If the regular administrator is absent for any length of time, another user can be moved into the administrative Organizational Role temporarily to manage the network.

You create the Organizational Role object and assign specific rights depending on the characteristics needed for the role. You then assign users to the Organizational Role as occupants through NetWare Administrator or NETADMIN.

Summary

Efficient planning decrease the amount of time necessary to manage installation of NetWare 4 by placing users, services and resources in proximity to each other within the tree.

This allows you to grant most rights to a container and have those rights flow through the tree to the users that will need them. For example, applying rights once to a container could effectively manage all the

resources within a given container, which minimizes the time spent to administer the Directory tree and reduces network traffic.

Evaluation

Once you have completed your accessibility plan, review the following questions to evaluate the efficiency of your plan:

- ◆ What applications are used in the organization?
- ◆ What applications are used by everyone on the network?
- ◆ Are any resources such as applications, directories, or printers shared across different locations or workgroups?
- ◆ Will all users in a certain container have the same access to a resource?
- ◆ How will you modify user access to resources?
- ◆ What client operating systems exist on the network?
- ◆ How many applications or network resources require bindery services?

Defaults

A new user has enough rights to read all their own properties, but can view only group membership, network address, and default server for other users. The login script is the only explicit read property defined for container objects.

NetWare administrators need to assign an explicit trustee of a property before the property can be used or shared. For example, a print job configuration does not work if defined at the container level and not assigned for a specific user.

Default Trustee Assignments

The default trustee assignments set at installation are as follows:

Trustee	Directory NDS rights	File System rights
User object that created the NetWare Server object	Supervisor [S] right to NetWare Server object	
User object ADMIN	Supervisor [S] right to [Root] object and to the NetWare Server object that it creates	
[Root] object	Read property [R] right to each Volume object's Host Server Name property and Host Volume Name property	
[Public] object	Read property [R] right to Server's Network Address property	
Server volumes	Read property right to Login Script property of the container	Read [R] and File Scan [F] to SYS:PUBLIC Read [R] and File Scan [F] to SYS:DOC (optional) Read [R], File Scan [F], Create [C] Edit [E] to SYS:

User Object Rights at Creation

User objects inherit the following default rights when they are created:

Table 6-3
User Object Rights

Object Name	Explicit Trustee	Object Rights from PUBLIC	Property Rights for User
Organization or Organizational Unit	Container name	Browse	Login script Read
Directory Map	None	Browse	None
Group	[Root]	Browse	Members Read
Profile script	None	Browse	None
User script	Username	Browse	All properties Read

Default Object and Object Property Rights

The default NDS object and NDS object property rights for newly created objects are listed in the following table. These rights are shown as an ACL entry with the following format:

- ◆ Which object or object property has the rights
- ◆ Rights to which object or object property
- ◆ What are the default rights

The term [Entry Rights] means the rights to the object itself, while [All Attribute Rights] means rights to all attributes of the object. Values not in brackets (such as Network Address) are actual property names.

For example, the [Creator] of a Group object has Supervisor rights to the object's [Entry Rights], meaning that the creator has Supervisor object rights to the object.

The [Root] object has Read rights to the Member object property, meaning that every user can read the membership of the group object.

Object Name	Default Object Rights	Default Object Property Rights
AFP Server	[Creator], [Entry Rights], [S] [Self], [Entry Rights], [S]	[Public], Messaging Server, [R] [Public], Network Address, [R]
Alias	[Creator], [Entry Rights], [S]	
Audit File	[Creator], [Entry Rights], [S]	
Bindery Object	[Creator], [Entry Rights], [S]	
Bindery Queue	[Creator], [Entry Rights], [S]	[Root], [All Attribute Rights], [R]
Computer	[Creator], [Entry Rights], [S]	
Country	[Creator], [Entry Rights], [S]	
Directory Map	[Creator], [Entry Rights], [S]	
External Entity	[Creator], [Entry Rights], [S]	[Root], Member, [R]
Group	[Creator], [Entry Rights], [S]	[Root], Member, [R]
NetWare Server	[Creator], [Entry Rights], [S] [Self], [Entry Rights], [R]	[Public], Messaging Server, [R] [Public], Network Address, [R]
Organization	[Creator], [Entry Rights], [S]	

Object Name	Default Object Rights	Default Object Property Rights
Organizational Role	[Creator], [Entry Rights], [S]	
Organizational Unit	[Creator], [Entry Rights], [S]	[Self], Login Script, [R]
Print Server	[Creator], [Entry Rights], [S] [Self], [Entry Rights], [R]	[Public], Network Address, [R]
Printer	[Creator], [Entry Rights], [S]	
Profile	[Creator], [Entry Rights], [S]	
Queue	[Creator], [Entry Rights], [S]	[Root], [All Attribute Rights], [R]
User	[Creator], [Entry Rights], [S] [Root], [Entry Rights], [B]	[Public], Message Server, [R] [Root], Group Membership, [R] [Root], Network Address, [R] [Self], [All Attribute Rights], [R] [Self], Login Script, [R,W] [Self], Print Job Configuration, [R,W]
Volume	[Creator], [Entry Rights], [S]	[Root], Host Resource Name, [R, W] [Root], Host Server, [R]

For objects that are installed with NetWare 4, the [Creator] is the ADMIN User object.

Note: The ability for a user to create the object in the first place derives from the user having a Create right to the container in which the object is created.

When you create an object, the server optimizes the ACL to remove unnecessary entries. Typically, this means that the ACL entry [Creator], [Entry Rights], [S] is removed, since in most cases the creator of an object has the Supervisor rights to the container where the object is found, and hence has the Supervisor rights to the newly created object by inheritance.

If, however, the creator only had the Create rights to the container, then the ACL for the newly created object retains the [Creator], [Entry Rights], [S] entry, since the creator would not otherwise have any rights on the newly created object.

Thus, if you create an object and then set its Inherited Rights Filter, you may no longer have access to the object, even though the [Creator], [Entry Rights], [S] ACL entry would appear to give you such rights.

Warning: Effective rights can be derived from security equivalence and inheritance, as well as being directly assigned to a user. When assigning rights to any NDS object property, you should understand how effective rights are computed.

Do not make nonadministrative users security equivalent to any NDS server object such as NetWare Server, AFP Server, or Print Server.

You should never assign any rights to [Public] beyond what is assigned at default. Any user, whether they are logged in or not, is security equivalent to [Public]. If you want to allow all users access to a property, it is better to assign those rights to [Root] or to the container the users are in.

Where to Go from Here

To	Go to
Create a migration strategy for servers and workstation from a previous version of NetWare or other network operating system	Chapter 9, "Developing a Migration Strategy," on page 173
Create an implementation schedule	Chapter 10, "Creating an Implementation Schedule," on page 197

Designing a Data Protection Plan

All network environments benefit from creating an effective data protection plan. You should ensure that some measure is taken to protect your network data.

This chapter describes the process used for designing a data protection plan for your network. The following topics are discussed:

- ◆ “Establishing a Redundant Hardware System” on page 150
- ◆ “Ensuring Adequate Partitioning and Replication” on page 151
- ◆ “Developing a Backup and Restore Strategy” on page 152
- ◆ “Creating a Disaster Prevention and Recovery Plan” on page 157

Introduction

NetWare® 4™ networks maintain data in the file system and in the Directory database. The file system stores files and applications that are used by network users and resources. The Directory database stores information that is used to maintain and manage operation of the network, such as access to network resources, printing, and security.

To adequately protect both file system and Directory database information, you should ensure that the following provisions are included in your plan:

- ◆ Redundant hardware
- ◆ Partitioning and replication
- ◆ Backup and restore system
- ◆ Disaster prevention and recovery plan

When creating a data protection plan for your network, it is important to remember that the file system and Directory database information are maintained as separate systems. This means that when you create the most efficient data protection plan for your network, you should analyze what methods ensure the most efficient data protection for each individual system.

Objectives

You should identify backup and restore needs, data integrity needs, and fault tolerance needs.

This will enable you to update your existing backup and restore methods and develop a disaster recovery plan for your network to determine the level of data protection you need.

Prerequisites

- You should have copies of the following planning documents:
 - ◆ Backup plans
 - ◆ Disaster recovery plans
 - ◆ Location maps
 - ◆ Resource maps
 - ◆ Directory tree structure
 - ◆ Partitioning guidelines
- You should have completed your Directory tree design. See Chapter 3, “Designing the Directory Tree Structure,” on page 25 for more information.

Establishing a Redundant Hardware System

Although the reliability of computer systems has improved considerably over the last several years, hardware failures still can—and will—occur. Numerous technologies are available to provide more reliable data storage and better protection against hardware faults.

Many of these technologies involve hardware redundancy of some kind.

For example, RAID (redundant array of inexpensive disks) hard disk systems maintain data even if one of the disks fails.

NetWare's disk mirroring and duplexing feature also protects against hardware failures in the disk channel. NetWare SFT III takes fault tolerance a step further and allows two servers to be mirrored. If a failure occurs on one server, the other one continues to service the network without interruption. See "Install NetWare 4.2 SFT III" in *Installation and Upgrade* for more information.

Another possibility is to maintain a *standby* server to which you can attach your regular server's external disk subsystem in case the server experiences an internal hardware failure. Several third-party vendors offer standby-server products that allow for a hot copy of data from one system to another.

Ensuring Adequate Partitioning and Replication

In a network with more than one server, NDS provides online backup of the Directory database through the placement of partition replicas on multiple servers. For example, in a two-server network with one partition, the NDS database is protected by placing a replica of the partition on each server. If one replica is lost because of a hard disk failure or because volume SYS: is damaged, NDS remains intact due to the replica on the other server.

In the event of an unforeseen loss of a server or volume, you can restore lost Directory information through an active replica. Replication also allows a replica that becomes damaged to be either rebuilt or recreated from another replica.

See Chapter 4, "Determining a Partition and Replication Strategy," on page 67 for more information.

Replication is the first line of protection for the Directory database. However, it does not provide sufficient protection in a single-server environment, or if replicas do not exist, or if all replicas become damaged. It also does not provide any protection for the file system. For these reasons, a regular backup of your network information must be maintained.

Developing a Backup and Restore Strategy

While replication provides the fault tolerance needed to maintain a working Directory tree, you must also include a full, offline backup of the file system and Directory database in your data protection plan.

It is important to remember that the Directory database is a distributed system that is dependent upon files stored on multiple servers. You cannot back up and restore the Directory database files directly on a per-server basis as done in previous NetWare versions.

NetWare 4 provides a backup and restore solution for your network through Storage Management Services™ (SMSTM). Using Novell's SBACKUP utility or third-party applications, you should design a data protection plan to ensure that your file system and network data are protected from loss or disaster.

Third-Party Solutions

Third-party backup package vendors can also use SMS to enable their backup software to work on a NetWare network. Contact your Novell® authorized reseller for a list of backup solutions that have been certified by Novell Labs.

You can also receive Novell Labs Bulletins with a list of the most current certification list by accessing the Novell Labs faxback system.

To access the Novell Labs FaxBack Service, complete the following steps:

1. Within the continental United States, dial 1-800-414-LABS (1-800-414-5227).
2. Outside the continental United States, dial 1-801-861-5544.

Follow the directions provided on the phone. You are prompted to enter a document number and then a fax number to send the document to.

Compatibility

Novell has SMS TSA software available for NetWare 3.11 and 3.12 servers, NetWare 4 servers, Novell Directory Services, NetWare SQLTM, and DOS and Windows, and UnixWare* clients.

It is highly recommended that the backup product used to back up your NetWare 4 servers utilize Novell's TSANDS.NLM to back up NDS, and TSA410.NLM to back up the file system.

Backup applications designed for the NetWare 3™ environment may not handle NetWare 4 data correctly. (For information about the capabilities of specific backup applications, contact the application vendor.) For backing up and restoring NetWare 4, you should use applications which fully support the SMS architecture.

Note: Because implementation details vary from vendor to vendor, it is important to review the documentation and readme files included with the SMS backup application of your choice. If the solution you choose has been tested and approved by Novell Labs, you can also obtain information on this product from Novell Labs Bulletins.

Determining Backup Administration Strategy

Since NDS is a distributed database, there can be multiple administrators who assist in maintaining the tree. In dividing up this responsibility, consider your backup needs and grant the necessary rights to those users who will perform the backup and restore operations.

In many cases, it works well to have a central backup administrator who has ultimate responsibility and a global view of the entire Directory tree for backup and restore purposes. Having a central backup administrator will help eliminate problems with incomplete backups resulting from the backup user not having sufficient rights to certain portions of the Directory tree.

Considering Network-Related Issues

Novell has identified four representative network environments, each of which has different ramifications on backup and restore strategies.

Single Server/Single Partition

In a single-server network, you can't have more than one replica. Further replication is not possible because there is no other server on which to store a replica. In this environment, you absolutely must maintain a full offline backup of the Directory database information. This is the only way you'll be able to restore lost Directory database data.

Multiple Servers/Single Partition

In a network with more than one server holding a single Directory partition, you should have at least two replicas of the partition, preferably three.

Maintaining an offline backup of the Directory database is also essential. If the servers are all located at the same site, be prepared in the event of a disaster that could destroy all the servers. Store a complete set of your backup media at another location, if possible.

Multiple Servers/Multiple Partitions

A network with multiple servers holding multiple partitions, both online replication and offline Directory database backup are essential. Wherever possible, have at least three replicas of each partition located on servers throughout the network.

Avoid having all replicas of a partition located on servers at the same site. It is a good idea to keep a complete set of your backup media off-site as well. Because SMS backup ignores partition boundaries, restoring the Directory tree, especially a partial restoration, is more involved. You'll need to keep a written record of how your partitions and replicas are set up.

Note: If you lose one partition and all its replicas, you can restore the objects that existed in that partition. However, the procedure for rebuilding the links to the lost portion of the Directory tree requires significant technical expertise and

should be performed with the assistance of a qualified technician. See "Multiple Servers/WAN Connections" for more information.

Multiple Servers/WAN Connections

Remote sites connected via WAN links require additional considerations for backing up the Directory tree. In particular, you should consider backup and restore needs when deciding how to administer the tree, either centralized with a single administrative user account that has full rights to the entire tree, or distributed with separate administrative accounts at each site.

To minimize data traffic across WAN links and to speed up backup and restore operations, consider performing backups locally at the remote sites. You might also design your Directory tree so that you have replicas of every partition stored at the same site as the backup host server. Then you can back up and restore the Directory database without having data go across WAN links.

If you back up and restore the Directory database and file system data for remote sites from a central location, you should ensure that the WAN connections, such as routers, bridges, and telecommunications links are functional before you begin.

If your host and target servers or workstations are not able to communicate across the network, a complete backup or restore is not possible.

Data Protection Guidelines

Efficient planning decreases the amount of time necessary to manage backup and restoration of the Directory database. It also ensures that a sufficient data protection plan is established before implementing NetWare 4 on your network.

The following guidelines should be followed when designing a data protection plan:

1. Use replication as the first level of protection for NDS.

In multiple server networks, have at least three replicas of every NDS partition stored on various servers throughout the network.

Use your offline SMS backup to restore NDS data only if it cannot be restored from a replica.

2. Choose a backup product that is certified for NetWare 4.

To handle NDS and other NetWare 4-specific features, the third-party backup program you use should be certified by Novell Labs. It should also support SMS and its TSA software. (For product certification information, call Novell Labs FaxBack at 800-414-5227 or 801-861-2776.)

3. Make sure you have the latest versions of backup-related software.

Periodically check with Novell and with your third-party vendor to ensure you have the latest version of the backup program, device drivers, TSA software, and so on. Novell provides updates on NetWire[®], the World Wide Web, and the NSEPro[™] CD-ROM.

4. Stay current with the newest operating system patches and NLM versions.

Periodically check Novell's electronic distribution sources (NetWire, World Wide Web, and NSEPro CD-ROM) for updates to the NetWare 4 operating system, Novell Directory Services (DS.NLM), and NDS utilities such as NDS Manager, DSTRACE, and DSREPAIR.

5. Keep a record of where NDS partitions and replicas are located.

Restoring NDS is much smoother if you have a record of your partitions and replicas. Be sure to note the full name of the container each server is added into and which replicas are stored on the server. See "Replica Placement Worksheet" for more information.

You can also use NDS Manager or DSREPAIR to record this information to a log file.

6. Ensure that NDS is fully functional and WAN links are up before backing up or restoring.

WAN links should be up and servers should be able to communicate with each other. All NDS partitions should be synchronizing without errors.

7. Back up NDS regularly.

The frequency of backing up NDS depends on how often you make changes to your tree. For trees that change often, back up NDS every time you do a full network backup. Always back up NDS before making major modifications to the tree.

8. Verify the completeness of each backup and restore session. After each backup and restore session, check the appropriate error and log files to make sure the process completed successfully and didn't skip crucial portions of your data.

9. Don't change NDS object names and contexts when restoring.

Don't use a restore situation to redesign your NDS tree. The restore process will go much more smoothly if you keep the NDS tree the same and restore servers to the same container objects as they were in before.

10. Always restore NDS information before restoring file system information.

File system trustee assignments will be affected by restoring NDS objects. To avoid problems, always restore NDS information before the file system.

11. Remember to re-update your software when reinstalling servers.

After reinstalling NetWare 4 or NDS from the original media, remember to reapply OS patches and recopy updated drivers, NLM software, and utilities before proceeding with a restore.

Creating a Disaster Prevention and Recovery Plan

A disaster prevention and recovery plan provides both preventative and responsive action to system or Directory failure caused by hardware failure in a NetWare server or by corruption in a Directory partition.

Writing and testing efficient disaster recovery plans and procedures is essential for recovery from catastrophic failures such as fire, floods, and earthquakes. For information on disaster recovery plans, see the manufacturer's documentation for your backup system or contact your Novell support provider.

Summary

Efficient planning decreases the amount of time necessary to manage backup and restoration of the Directory database. It also ensures that a sufficient data protection plan is established before implementing NetWare 4 on your network.

In multiple server networks, have at least three replicas of every NDS partition stored on various servers throughout the network. Use your offline SMS backup to restore NDS data only if it cannot be restored from a replica. Use replication as the first level of protection for NDS.

You should always keep your backup software current and ensure that it is compatible with all of the NetWare operating systems running in your network.

Where to Go from Here

To	Go to
Create a migration strategy for servers and workstation from a previous version of NetWare or other network operating system	Chapter 9, "Developing a Migration Strategy," on page 173
Create an implementation schedule	Chapter 10, "Creating an Implementation Schedule," on page 197

Designing an Application Management Strategy

All network environments benefit from creating an effective application management strategy.

This chapter describes the process designing an application management strategy for your network. The following topics are discussed:

- ◆ “Ensuring NetWare Compatibility” on page 161
- ◆ “Creating Efficient and Intuitive Application Directories” on page 162
- ◆ “Identifying Efficient Application Management Tools” on page 165
- ◆ “Identifying Efficient Licensing and Metering Tools” on page 168
- ◆ “Application Management Guidelines” on page 170

Introduction

Network applications are the productivity tools for your organization. How you manage these tools depends on the type of applications you are using.

Network applications can be divided into three types:

- ◆ Network-aware
Applications that run efficiently on the network, but do not use any special network features such as storage and messaging services.

- ◆ Network-enabled

Applications that run efficiently on the network, but use proprietary solutions for services such as authentication, print management, and messaging.

- ◆ Network-integrated

Applications that are designed to take advantage of special network features such as the Directory database and storage services.

Most network applications in use today are network-aware, but not truly network-integrated. This poses a challenge for network administrators when managing large databases across multiple applications and maintaining proprietary methods for implementing network services such as messaging and printing.

Following some simple guidelines and using efficient management tools can greatly reduce the amount of time and effort needed to manage applications.

Objectives

The objectives in designing an application management strategy are

- ◆ Ensuring that all your applications are compatible with the version of NetWare[®] you are using.
- ◆ Creating an efficient and intuitive directory structure for the application software and support components.
- ◆ Identifying tools that provide ease of distribution, installation, access, and operational control.
- ◆ Determining and implementing an efficient licensing and metering strategy

Prerequisites

- ❑ You should have copies of the following planning documents:
 - ◆ Backup plans
 - ◆ Location maps
 - ◆ Resource maps
 - ◆ Directory tree structure
 - ◆ Partitioning guidelines
- ❑ You should have completed your Directory tree design. See Chapter 3, “Designing the Directory Tree Structure,” on page 25 for more information.

Ensuring NetWare Compatibility

You should determine whether your applications are NetWare compatible before you purchase them. Approximately 5,000 software packages are compatible and registered with Novell[®]. This compatibility information is important because NetWare makes demands on application software that can cause corrupt data or impede user's productivity.

Information about NetWare compatibility and registration are available on NetWare[®] or from your local Novell sales office. You can also contact the software vendor directly.

Novell's Yes Program

The Yes Program, Novell's trademarking and certification program, helps Novell customers identify and purchase Novell-compatible third-party hardware and software products. The Yes Program provides developers with the opportunity to test and certify their products against Novell's strict quality standards to ensure that their products are compatible with Novell products. Once products have passed the Yes Program's business and certification requirements, they are eligible to use a Yes logo.

The Yes Program identifies products that are compatible with NetWare network operating systems and other Novell products such as ManageWise™, GroupWise™, and NetWare Telephony Services™.

Ensuring Applications Are Designed for Multiple-User Access

All applications should support multiple users simultaneously. This means that it provides file-sharing and multiple-user access.

If your applications are designed to run on standalone computers, don't assume that they will automatically operate on the network. NetWare 4™ allows almost all single-user applications to run across the network, but, it does not provide data or resource sharing for these applications.

Using the Application Compatibility Template

Before installing and configuring applications on NetWare 4 in your network environment, test the applications in a lab environment. For information about setting up a lab, see "Setting Up a Lab" on page 191.

While testing the applications in the lab, use the Application Compatibility template to record your test results. For a copy of the template, see "Application Compatibility" on page 252.

Creating Efficient and Intuitive Application Directories

Creating efficient and intuitive application directories requires you to

- ◆ Create the application directory structure
- ◆ Provide adequate load balancing
- ◆ Protect the application directories

Creating the Application Directory Structure

Before installing and configuring applications on a NetWare server, you should create an efficient and intuitive directory structure to install the applications in. This also provides for easy and efficient access control implementation.

Your application directory structure should be designed to meet the specific needs of your organization and applications requirements. Most organizations have a variety of applications that are supported on many operating systems. The structure you choose should allow for access by all types of client workstations.

The most important consideration to make when creating application directories is to separate the applications program files from the data files that are created within the applications.

By separating the program files from the data files, you can easily control access and better support data protection plans. You might also consider separating program and data files by placing each on different volumes or servers. This will help balance the load on servers by reducing the amount of traffic on the server.

Providing Adequate Load Balancing

To ensure that adequate load balancing is provided, consider the following:

- ◆ Sufficient hardware exists to support the number of necessary connections, RAM and disk space requirements, and processor speed.
- ◆ Applications servers are physically near the users that use its applications.
- ◆ There are enough servers to efficiently support the number of users who need access to the applications.
- ◆ Access control is established to ensure that access to application server is spread equally across servers.
- ◆ Application licenses that might be server-based require additional licenses for more than one server. Ensure that you have enough licenses to support your organization's needs.

By making considerations for load balancing, you can greatly enhance your network's flexibility and potential for optimization. You can also configure management of the application servers to support centralized or distributed methods to suit your management style.

Protecting Application Directories

Your application programs must be protected from a variety of conditions or disasters.

You should ensure that you are protected from the following conditions through proper access control:

- ◆ Accidental deletion
- ◆ Intentional deletion
- ◆ User access
- ◆ Concurrent use of application licenses

For information about setting up and configuring proper access control, see Chapter 6, “Creating an Accessibility Plan,” on page 107.

You should ensure that you are protected from the following conditions through proper disaster recovery methods and backups:

- ◆ Server failure
- ◆ Catastrophe
- ◆ Application or data theft
- ◆ Software updates

For information about setting up and configuring a sufficient data protection plan, see Chapter 7, “Designing a Data Protection Plan,” on page 149.

Identifying Efficient Application Management Tools

Application management consists of managing ease of distribution, installation, access, and operational control.

There are many tools to assist you in managing applications on the network. The solution that you chose should provide at least the following functionality:

- ◆ Distribution

The ability to successfully distribute applications from a server to multiple clients.

- ◆ Installation

The ability to perform necessary configuration management for each client an application is deployed to.

- ◆ Operational control

The ability to execute required control tasks against an application; for example, application startup and shutdown, or network drive connection.

Generally speaking, the solution you choose must enhance your ability to perform the basic tasks required to manage the entire lifecycle of a network application.

Using NetWare Application Management Tools

NetWare 4.2 provides application management software that helps you set up and manage network applications from a single administrative console through Novell Directory Services™ (NDS™).

The software is comprised of a NetWare Administrator tool referred to as NetWare Application Manager™ (NAM™) and a client tool referred to as NetWare Application Launcher™ (NAL™).

The NetWare application management tools enables management of the application lifecycle, beginning with distribution and installation to configuration and upgrades.

Using NetWare Application Manager

The NetWare Application Manager (NAM) software assists you in setting up and managing network applications from the NetWare Administrator.

Using NetWare Administrator, an administrator creates Application objects in NDS for any application to make it available through the NetWare Application Launcher (NAL). Rights to these application objects can then be assigned to containers, groups and users.

Using the NetWare Application Launcher

The NetWare Application Launcher (NAL) is a user application that leverages NDS to offer users easy access to applications stored on the network.

NAL offers easy distribution, updating, version control and license management for applications stored on the network. NAL also offers fault tolerance for the application environment.

When a user running NAL logs in through Novell Directory Services, NAL looks at the groups and containers the user belongs to and recognizes any applications that the user is authorized to access.

In the background, NetWare Application Launcher locates all the user's applications on the network and accesses them transparently when the user selects the program icon in Windows. An NAL program group displays all the appropriate application icons that the user can select to launch the application.

Available applications are scanned and updated automatically for the user.

Users don't need to worry about drive mappings, paths, or rights to the application directories. The administrator can manage the application launcher on a Container, Group, or User object level.

Understanding the Benefits of NetWare Application Management Tools

The NetWare Application Manager and NetWare Application Launcher

1. Simplify administration of network applications

The user cannot delete the application icon or change any of its information.

2. Eliminate the need for login scripts

The administrator can associate working directories, paths to executable files, or other information with the Application object.

3. Simplify user access to network applications

The users can log in to the network using any workstation and can still access all their applications.

4. Dynamically update user desktops

The users are unaware of any changes if the administrator wants to move or rename the executable file. Only the object itself must be modified.

5. Ease installation and upgrades

Users can run installation programs from the NetWare Application Launcher, allowing them to install software such as Novell Client™ Software, productivity software, and operating systems over the network. An application can be upgraded by modifying the path to the new application executable, which can be installed anywhere on the network.

6. Easy to install and use

Steps for installing the NetWare Application Manager are easy to follow and to complete. Launching the NetWare Application Launcher is as easy as double-clicking an icon.

For information on how best to use NAL and NAM in your network, see the online help provided through the NetWare Administrator help menu or see “Setting Up and Using NetWare Application Management Software” in *Supervising the Network* .

Identifying Efficient Licensing and Metering Tools

NetWare 4 also provides NetWare Licensing Services (NLS) that enables you to monitor and control the use of licensed applications on a network.

Metering Applications

Application metering software controls the number of people who can simultaneously access a network application. This type of software helps you use an application within the limits of the software license.

Licensing Applications

Almost all legitimate computer software use is regulated by an explicit license. The license typically states who may use the software and under what conditions. There are many different types of licenses, each of which reflects the intended use of the software.

Until recently, software use licenses were often nothing more than a printed license statement included in the product's packaging. Software vendors relied on the integrity of their customers to not violate the license; in many cases, this was sufficient to protect the vendor's investment in developing the software.

However, in an attempt to further reduce losses that result from illegal software distribution and use, a group of software developers have written the License Service Application Programming Interface (LS API). The License Service API allows license servers to communicate with client applications via the API making applications essentially self-metering.

Key to the automatic enforcement of license agreements is a third component called the access token, or digital license certificate. The LS API contains the specification of the industry standard License Service API, or LS API for this component.

Using NetWare Licensing Service

NetWare Licensing Service (NLS) is the means provided by Novell by which applications that are written to the LSAPI specification can be managed in a NetWare environment.

NetWare 4 provides NLS.NLM which offers a standard server component for developers to write to. The NLS technology provides more active enforcement of application license agreements. NetWare 4 can now function as a licensing server giving developers more control over how applications are being used.

To take advantage of NLS, the software on your network must incorporate the LSAPI specification. For information on whether the software you use is written to the LSAPI specification, contact the appropriate software vendor.

Managing Application Licenses

NetWare Licensing Service is a distributed, enterprise network service that enables administrators to monitor and control the use of licensed applications on a network.

NLS is tightly integrated with Novell Directory Services (NDS) and is based on an enterprise service architecture. This architecture consists of client components that support different platforms and system components that reside on NetWare 4 servers.

Metering Application Use

NLS also provides a basic license metering tool, as well as libraries that export licensing service functionality to developers of other licensing systems.

NLS Components

NLS also provides a basic license metering tool, as well as libraries that export licensing service functionality to developers of other licensing systems.

NLS consists of the following components:

- ◆ One or more License Service Providers loaded on NetWare 4 servers. An LSP server is a NetWare 4 server with the NLS NetWare Loadable Module™ (NLS.NLM) loaded.
- ◆ By default, when you install NLS, an LSP Server object is created in the Directory that represents the LSP server. The LSP Server object is placed in the same context as the NetWare Server object that represents the NetWare 4 server on which you installed the NLM.
- ◆ Platform-specific client components NLS supports DOS, Windows*, Windows 95/98, Windows NT*, and NetWare 4 NLM clients
- ◆ Novell Directory Services (NDS)
- ◆ Transaction databases

For information on how best to use NLS in your network, see the online help provided through the NetWare Administrator help menu and “Managing NetWare Licensing Services” in *Supervising the Network*.

Application Management Guidelines

The following guidelines should be followed when designing a application management strategy:

1. Install all applications under the same username, such as the ADMIN User object. This keeps ownership consistent and manageable. Be aware that there are some applications that check for the SUPERVISOR user, not an equivalent.
2. Observe the license restrictions on all software.
3. Take advantage of UNC (Universal Naming Convention) path names and Novell Directory Services. By using UNC path names, workstations reference the server and volume, not a drive letter. This reduces the number of drive connections and enables you to change server names by leaving and Alias object for the server in the Directory tree.

4. Use the MAP ROOT command to support programs that require a root directory.
5. Use Directory Map objects to ease access to programs and applications.
6. Use NetWare Application Manager and NetWare Application Launcher to ease distribution, installation, and access to network applications.
7. Ensure that application directories have efficient access control. Assign Read and Files Scan rights to application groups or users. Use the FLAG utility to assign Read-Only and Shareable attributes to program files.

Summary

Efficient planning decreases the amount of time necessary to manage application installation, distribution, and configuration.

Using some simple guidelines and efficient management tools can greatly reduce the amount of time and effort needed to manage applications.

Where to Go from Here

If you want to	Go to
Create a migration strategy for servers and workstation from a previous version of NetWare or other network operating system	Chapter 9, "Developing a Migration Strategy," on page 173
Create an implementation schedule	Chapter 10, "Creating an Implementation Schedule," on page 197

Developing a Migration Strategy

Existing NetWare[®] networks and networks based on non-NetWare operating systems should develop a strategy to ensure an efficient and trouble-free migration of network data to NetWare 4[™]. This chapter describes the process used for developing a migration strategy for your network.

The following topics are discussed on the indicated pages:

- ◆ “Determining a Client Migration Method” on page 175
- ◆ “Determining a Server Migration Method” on page 181
- ◆ “Setting Up a Lab” on page 191

If you are installing a new network, you do not need to develop a migration strategy. Continue to the next procedure. See Chapter 10, “Creating an Implementation Schedule,” on page 197 for more information.

You might want to review information about setting up a test lab for integration testing of hardware and software applications if your network environment meets any of the following requirements:

- ◆ More than thirty NetWare 4 servers
- ◆ More than 2000 users

See “Setting Up a Lab” on page 191 for more information.

Introduction

Creating an efficient migration strategy is important to the success of your NetWare 4 implementation. By doing so, you can successfully transition existing network resource settings and data to NetWare 4.

The transition between previous versions of NetWare and other network operating systems to NetWare 4 is simple and automatic. Tools are provided to assist you.

An efficient migration strategy requires you to develop strategies for

- ◆ Client workstation migration
- ◆ Server migration

Other factors that will affect the success of your implementation should be managed through lab testing and setting up a pilot system. These factors are:

- ◆ Software compatibility
- ◆ Hardware compatibility

By testing these factors in a lab setting, the team will have more time to familiarize themselves with the new operating system and utilities.

Objectives

You should develop an efficient strategy for migrating client workstations and network servers, select the best migration method for your particular network environment, and then test the compatibility of network software and hardware by setting up a test lab and pilot system.

Use resource maps, location maps, LAN and WAN topology maps, installation and configuration information, backup schedules, and workflow information to determine the best strategy to use to for migrating NetWare 4.

Prerequisites

- You should have copies of the following documents:
 - ◆ Physical maps
 - ◆ Logical maps
 - ◆ Installation information
 - ◆ Configuration information
 - ◆ Backup schedule
 - ◆ Workflow information

Determining a Client Migration Method

NetWare 4 supports the following client types:

- ◆ DOS/Windows, Windows 95/98, and Windows NT
- ◆ NFS* UNIX*

NetWare 4 provides full NDS support for DOS and Windows clients running the 16-bit NetWare DOS Requester™ software and the Novell Client™ software. The NetWare DOS Requester supports all of the migration and administration utilities.

NetWare 4 provides full bindery-based login for NFS clients. All resources are accessed through bindery services.

Migrating Client Software before Installing NetWare 4

You should migrate all client workstations to NetWare 4 before migrating NetWare server platforms. If you are migrating client workstations connected to non-NetWare servers, wait until the server data is migrated before installing the Novell Client software on those workstations.

Identifying Critical Factors

Consider the following when migrating client workstations:

Topic	Condition	Decision
Protocols	The protocols and frame types of your network affect the client software configuration.	NetWare 4 supports the default frame type for Ethernet_802.2 as opposed to the previous setting of Ethernet_802.3. NetWare 4 also supports NetWare/IP™ and IPX™ protocols running simultaneously or separately. Decide which frame types and network protocols will be used.
User context	A default context is set in the NET.CFG file for DOS and Windows workstations.	Novell Clients for DOS and Windows support the NAME CONTEXT setting in the NET.CFG file. This allows workstations to set the default setting for the user context when they log in to the network.
Workstation management	The management software used affects the client configuration.	The Novell Client supports SNMP-based management tools. Decide if you will load the Novell Client support for SNMP.
Workstation backup	The backup software used on the network affects the client configuration.	The Novell Client supports target services agents (TSA) for SMS-based backup engines. Decide if you will load the Novell Client support for SMSTSA.
Security	Networks that require a high level of security might need additional configuration.	The Novell Client supports RSA encryption and packet signature functionality. Decide if your network requires this level of security.
Backwards compatibility	Some network software and hardware were developed for bindery services.	Novell Client software supports connectivity to bindery-based servers and applications.

Topic	Condition	Decision
Coexistence with peer-to-peer networks	Some client workstations are required to connect to other networks.	Novell Client supports connectivity to Personal NetWare™ and NDIS*-based networks. Decide if DOS and Windows workstations will support connectivity to Personal NetWare and other NDIS-based networks.
Performance	All workstations benefit from performance tuning the client software.	Novell Client software allows for Large Internet Packets (LIP) and Packet Burst™. Decide if all workstations require the performance increase.

Novell provides the following client software for the different types of workstations:

- ◆ Novell Client for Windows 95/98
- ◆ Novell Client for DOS and Windows 3.1x
- ◆ Novell Client for Windows NT 4.0

NetWare NFS Services is provided in NetWare 4. This allows UNIX client workstations bi-directional access to file and print services between UNIX and NetWare Servers.

Client Type	Description
Novell Client software	<p data-bbox="598 208 1212 377">Novell's clients are based on a common, advanced architecture that departs from the NetWare DOS Requester software. This new architecture enables the client software to run in protected mode. In addition, the Novell Client software requires less than 5 KB of conventional memory, while providing a larger cache.</p> <p data-bbox="598 407 1212 483">The Novell Client architecture, designed for robust connectivity and easy maintenance, provides the following features:</p> <ul data-bbox="598 513 1212 1478" style="list-style-type: none"> <li data-bbox="598 513 1212 589">◆ You can distribute the Novell Client software to the computers on your network using the Automatic Client Upgrade utility. <li data-bbox="598 619 1212 818">◆ The Novell Client software detects changes in a workstation's network environment and restores connections to the network when the relevant network service is restored. This makes the Novell Client software the most reliable Novell Client available. And, when a computer loses its connection to the network, the computer continues to run without having to reboot. <li data-bbox="598 848 1212 959">◆ The Novell Client software caches frequently used data, such as file content and network information, resulting in less traffic on the network and faster response times on the client. <li data-bbox="598 989 1212 1065">◆ The Novell Client software supports multiple Directory tree access and complete Novell Directory Services access. <li data-bbox="598 1095 1212 1478">◆ The Novell Client software can use 32-bit or 16-bit LAN drivers. Client 32 supports the following LAN drivers: <ol data-bbox="598 1178 1212 1478" style="list-style-type: none"> <li data-bbox="598 1178 1212 1289">1. 32-bit ODI LAN drivers that comply with the latest NetWare driver specifications (Some certified drivers for NetWare 4 are compatible with the Novell Client software.) <li data-bbox="598 1319 1212 1365">2. 16-bit ODI LAN drivers that comply with the latest NetWare driver specifications. <li data-bbox="598 1395 1212 1478">3. 32-bit and 16-bit Network Device Interface Specification (NDIS) adapter drivers (Novell Client for Windows 95/98 only)

Client Type	Description
Novell Client for DOS and Windows 3.1x	<p>With NetWare 4, Novell introduced a new DOS client architecture called the NetWare DOS Requester software.</p> <p>The NetWare DOS Requester software consists of individual functions in NET.CFG for the client.</p> <p>With NetWare 4, the NetWare DOS Requester software has been enhanced in the following ways:</p> <ul style="list-style-type: none">◆ Using the Automatic Client Upgrade feature, you can easily upgrade workstations using the NetWare DOS Requester software to the new Novell Client for DOS and Windows software.◆ With the 16-bit version of the NetWare Application Launcher™ utility, workstations using the NetWare DOS Requester™ software can access Application objects in the Directory.◆ A version of the NetWare DOS Requester software that includes support for a 16-bit TCP/IP transport, the Dynamic Host Configuration Protocol (DHCP), and the NetWare/IP software is included with NetWare 4.

Client Type	Description
Novell Client for Windows NT 4.0	<p>The Novell Client for Windows NT allows Windows NT workstations to integrate into NetWare networks. Users can log in to their NT workstation once and access all the services on their NetWare network.</p> <p>The NT client software includes support for NDS, enabling NT client workstations to take advantage of all of features of NDS, including transparent access to all the resources on the network, regardless of where they are physically located. The product runs on Windows NT 4.0 and includes the following features:</p> <ul style="list-style-type: none"> ◆ Full NDS support ◆ Access to NetWare file and print services through NT File Manager and NT Print Manager ◆ Support for NDIS 32-bit server LAN drivers (all NetWare 4.1 certified server LAN drivers are supported) ◆ Support for both IPX and NetWare IP transport protocols for accessing NetWare services ◆ Compatibility with any DOS or Win16 application for backward-compatibility ◆ NetWare support for NT long file naming through Novell's HPFS name space

Automating the Client Migration Process

You can automate the migration process for existing Novell Client workstations by using the Automatic Client Upgrade (ACU) instead of the INSTALL.EXE program.

Using Automatic Client Upgrade (ACU)

With NetWare 4, you can use the Automatic Client Upgrade (ACU) feature to easily migrate a set of network clients. The following table summarizes the upgrade scenarios that ACU is designed to address.

From.	To
Workstations using the 16-bit NetWare DOS Requester software	The Novell Client for DOS and Windows software
Workstations using an outdated version of the Novell Client for Windows 95/98 and Windows NT software	The latest version of the Novell Client for Windows 95/98 and Windows NT software
Workstations using a version of Microsoft's Client for NetWare Networks	The latest version of the Novell Client for Windows 95/98 and Windows NT software

For complete information on using the Automatic Client Upgrade feature, see the help system for the Novell Client for DOS and Windows or Novell Client for Windows 95/98 and Windows NT software (depending on the software to which you want to upgrade).

Determining a Server Migration Method

Novell[®] supplies various options for upgrading earlier versions of the NetWare operating system (NetWare 3 and 4) to NetWare 4.2. The upgrade option you use is dependent on a number of variables which include

- ◆ Your present version of NetWare
- ◆ Your hardware, including servers and clients
- ◆ Your intention of maintaining an existing server, or migrating bindery and data to a new server

Identifying Upgrade Methods

There are three ways to migrate to NetWare 4:

- ◆ Upgrade Using INSTALL.NLM

This option allows you to maintain the NetWare 3.1x or 4.x computer as a server by upgrading the operating system to NetWare 4.

If you are upgrading from a NetWare 3.1x server, the server is placed in a context within a NetWare 4 Directory tree.

NetWare 4 requires a volume SYS: of at least 75 MB. If your NetWare 3.1x or 4 server's volume SYS: is smaller than 75 MB, and you want to maintain the computer as a server, you must perform a Same-Server Migration.

- ◆ Upgrade Across-the-Wire Using INSTALL.NLM or the Novell Upgrade Wizard

This option is for those who have an existing NetWare 3.1x server and want to migrate the server bindery and data across-the-wire to an existing NetWare 4 server using the Novell Upgrade Wizard.

This option allows you to see and refine a model of your updated Directory tree before completing its migration.

Then, you can migrate the server files using the NetWare File Migration utility (for NetWare 3.1x servers).

The Novell Upgrade Wizard

The Novell Upgrade Wizard provides a powerful yet easy-to-use tool that makes migrating from NetWare 3 to NetWare 4 or 5 simple and cost effective. The Novell Upgrade Wizard steps you through the migration process. It is as easy as dragging and dropping objects from one location to another. This wizard migrates both the NetWare 3 bindery and the file system to an existing NetWare 4 or 5 network. All bindery objects are upgraded to NDS objects.

The Novell Upgrade Wizard includes a variety of innovative and unique features that equal or surpass other migration tools. Some of these features include:

- ◆ Drag and drop modeling capability
- ◆ Ability to migrate both the NetWare 3 bindery and file system in a single utility
- ◆ Password migration functionality
- ◆ Option to establish migrated users' rights based on an existing user template

- ◆ Option to create new user templates for migrating users
- ◆ Ability to create new containers and subdirectories in the existing NDS™ tree
- ◆ Ability to browse the entire NDS tree
- ◆ Ability to run the utility on its own, or as a snap-in to NetWare Administrator
- ◆ Ability to create login scripts
- ◆ Notification of possible errors and ability to correct these errors before migration
- ◆ Progress bar indicating percent completion
- ◆ Ability to run on both Windows 95/98 and Windows NT workstations

See the *Installation and Upgrade* for information on how to load and use the Novell Upgrade Wizard.

Identifying Other Upgrade Options

The options discussed previously are the principal options used to upgrade to the NetWare 4 operating system. Novell continues to provide and support other network operating system upgrade options. These include:

- ◆ A solution for upgrading NetWare 3 and 4 LANs and WANs using RCONSOLE.
- ◆ The UIMPORT utility. This is provided to allow you to import Directory information from another database to Novell Directory Services™.
- ◆ Tools for migrating non-NetWare operating systems to NetWare 4 (available through Novell Consulting Services).
- ◆ Additional upgrading tips and ideas (available through Novell Consulting Services).

Maintaining Backwards Compatibility

You should upgrade all NetWare 4 servers to NetWare 4.2 for performance and administrative advantages.

However, during migration to NetWare 4.2, different versions of NetWare 4 and NetWare 3 will still interoperate. NetWare 4 networks support this by offering bindery services and NetSync.

Maintaining Bindery Services in a NetWare 4 Environment

Some applications, services, and clients that run in the NetWare 4 environment do not currently take full advantage of Novell Directory Services technology. To enable users of these services to access them from the NetWare 4 environment, Novell offers bindery services.

With bindery services, NDS imitates a flat structure for leaf objects within an Organization or Organizational Unit object. Thus, when bindery services is enabled, all objects within the specified container can be accessed by NDS objects and by bindery-based servers and client workstations.

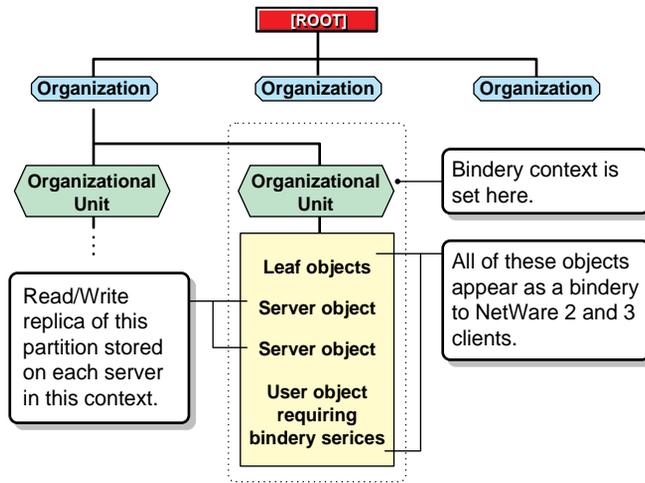
Important: Bindery services applies only to leaf objects in the specified container object.

Setting a Bindery Context

To enable bindery services, use the SET BINDERY CONTEXT=*complete_name* parameter by using the SET command or the SERVMAN server utility. (See “SERVMAN” in *Utilities Reference*.) The container object you indicate with the SET BINDERY CONTEXT parameter is called the *bindery context*.

The following figure illustrates bindery services when an Organizational Unit object is specified as the bindery context.

Figure 9-1
Bindery Services in a Directory Tree



A writable replica of the partition that includes the container object to be set as the bindery context must be stored on each server you want bindery services enabled on.

However, by default, only the first three servers installed on a partition receive a replica of the partition during the installation process and subsequently support bindery services.

You can add replicas to other servers if needed for bindery services. If a read/write or master replica is not present, use the NDS Manager or PARTMGR utility to add one to the server. For information and procedures, see “Placing Replicas for Bindery Services” on page 83.

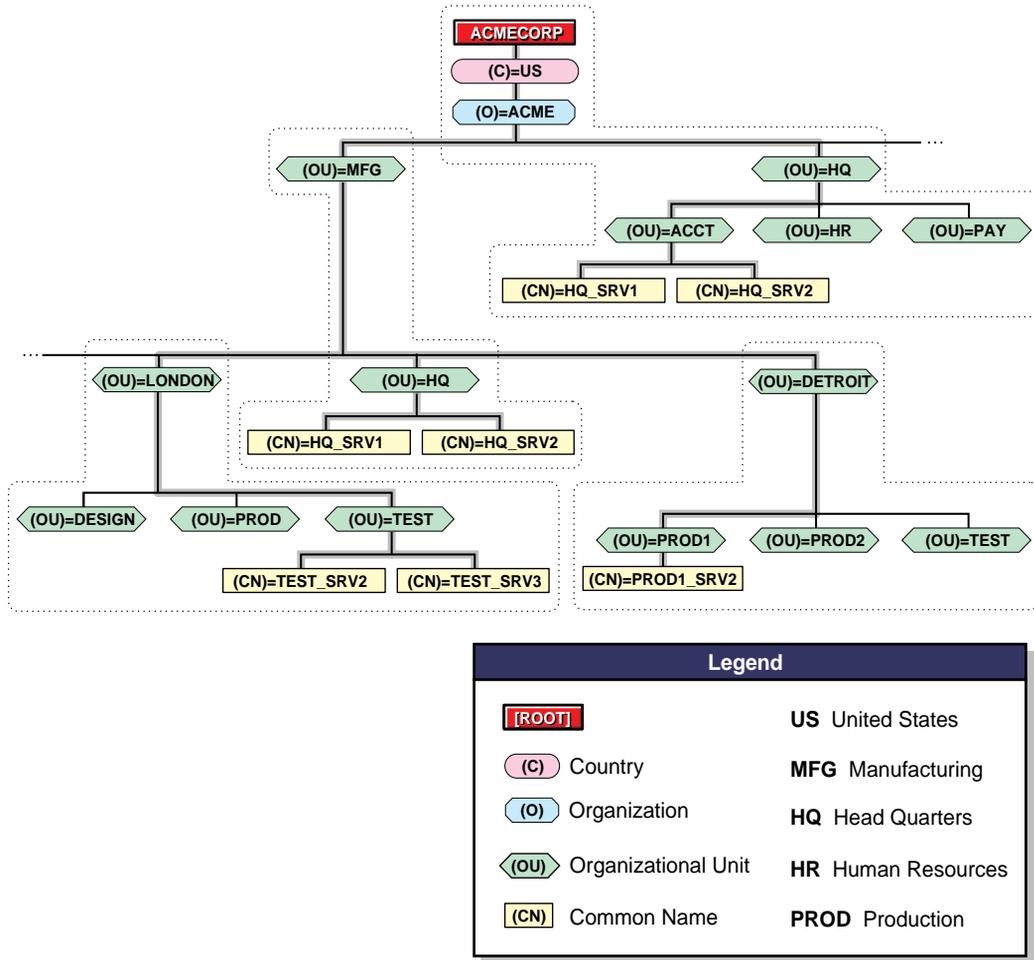
Important: If a bindery context is not set, Novell Directory Services (NDS) cannot support bindery services.

Setting Multiple Bindery Contexts

Ideally, all objects a user wants to access under bindery services should be located in the same bindery context. However, this is not always possible or practical.

You can set multiple bindery contexts for users who need to access objects outside of their own bindery contexts. For example, consider the Directory tree in the following figure.

Figure 9-2
Multiple Bindery Contexts



To set bindery contexts on the servers HQ_SRV1 and HQ_SRV2 in this figure, you would enter the following command in the server's AUTOEXEC.NCF file where the users are located:

```
SET BINDERY CONTEXT=ACCT.HQ.ACME;PROD1.  
DETROIT.MFG.ACME;TEST.DETROIT.MFG.ACME;
```

To set multiple bindery contexts, you must set the contexts to include the path all the way to the [Root] of the tree. You can set up to 16 contexts per server. Use semicolons to separate the complete names of the containers that you want a bindery context set to.

Warning: Do not change a server's bindery context once you set it. Changing a server's bindery context prevents all bindery services users (from the original context) who need to log in to that server from accessing bindery services. Changing the server's bindery context can also disable access to print queues.

Bindery services allows NetWare 4 servers to emulate earlier versions of NetWare and is, therefore, server-centric. For instance, if a client workstation requests a bindery login, bindery services directs the default server to use the bindery login script found in the user's mail directory on the SYS: volume instead of using the user's global NDS login script. Changes to the bindery login script are kept locally and are not distributed to other servers.

You cannot disable bindery services if someone is logged in via bindery services, and bindery objects are always available unless bindery services is disabled.

Planning Bindery Services

When you plan and implement bindery services, you need to consider the following.

Created Objects

Keep these guidelines in mind as you plan bindery services:

- ◆ If you require the user GUEST, or if you use a service that requires GUEST, you must create such a user in the Directory database.
- ◆ During installation, a bindery object SUPERVISOR is created but is not used with NDS. The NDS utilities do not display this object. This object is intended to be used with bindery services and to enable access to the server via a bindery login. Once bindery services is enabled, you can use this object to log in to the server, providing you log in as a bindery object. You can create an NDS User object SUPERVISOR and assign ADMIN-equivalent rights to it in NDS. However, the bindery object and the Directory object are unique and separate objects even though they are identified by the same name.

- ◆ After installing NetWare Services, you can use a migration utility to convert bindery user accounts to Directory User and Group objects. If you do, all users except SUPERVISOR and all groups are updated to Directory objects. The user SUPERVISOR is migrated, but with supervisory rights for that server's file system and bindery context only. The supervisor does not appear as an Directory object.

Inaccessible Information

Some NDS information is not available to users through bindery services. This information includes, but is not limited to, the following items:

- ◆ E-mail name
- ◆ Phone number
- ◆ Print job configurations
- ◆ Aliases
- ◆ Profiles
- ◆ NDS login scripts

Limited Partitioning

The bindery context for a server can be set to a container that is part of a partition stored on a different server. But, before you can use bindery services, you must place a writable replica of the partition that includes the bindery context on the bindery services-enabled server.

If you set the bindery context for a server to a container object that is not part of a writable replica on that server, users will not be able to log in via bindery services.

For more information, see “Placing Replicas for Bindery Services” on page 83.

Maintaining NetWare 3 Servers Using NetSync

NetSync synchronizes NetWare 3 users and groups with Directory objects in specific NetWare 4 server bindery contexts. When you update or create a user or group on the NetWare 4 server, the NetWare 4 server synchronizes the information with the NetWare 3 servers in the NetSync cluster. The user or group now exists on all the NetWare 3 servers in the cluster.

Setting up NetSync allows you to

- ◆ Upgrade NetWare 3 print services to NetWare 4
- ◆ Administer NetWare 3 servers, users, and groups with NetWare 4 utilities
- ◆ Maintain up to 12 NetWare 3 servers in a NetSync cluster
- ◆ Remove NetWare 3 servers from a NetWare Naming Service (NNS) domain for migrating to Novell Directory Services (NDS)

Determining When to Use NetSync

You should use NetSync for either of the following two reasons:

- ◆ You want to access and manage existing NetWare 3 users, groups, and print queues services from NetWare 4 and do not plan upgrading the NetWare 3 server.
- ◆ You want to remove a NetWare 3 server from a NetWare Naming Service (NNS) domain

Determining When to Not Use NetSync

You should not use NetSync for any of the following reasons:

- ◆ You only have NetWare 4 servers in your network
- ◆ You are upgrading all NetWare 3 servers to NetWare 4 and migrating the user and group accounts, and print services
- ◆ You plan to manage the NetWare 3 servers outside of your NetWare 4 network

For more information about setting up and using NetSync, see *Installing and Using NetSync*.

Maintaining a Mixed NetWare 4 Environment

You should upgrade all NetWare 4 servers to NetWare 4.2 for performance and administrative advantages. However, during migration to NetWare 4.2, different versions of NetWare 4 and NetWare 3 will still interoperate.

Maintaining a mixed NetWare 4 environment will require some special maintenance to ensure the correct version of NDS.NLM is being used. Furthermore, you will need to understand some specific partitioning limitations when distributing partitions across mixed NetWare 4 servers.

Checking Your Novell Directory Services Version

The NDS base schema has been modified in NetWare 4.2. The new schema is compatible with DS.NLM version 5.00 and later, and the DS.NLM versions supported in NetWare 4.2.

You can check which version of DS.NLM you are loading by going to the server console and typing

```
MODULES <Enter>
```

A sample might appear as follows:

```
DS.NLM
  NetWare 4.2 Directory Services
  Version 6.00  September 23, 1998
  Copyright 1993-1998 Novell, Inc. All rights
  reserved
```

The sample above indicates that the NetWare 4.2 server is using DS.NLM version 6.00.

If	Then
You are upgrading a NetWare 4.1 server that is running a DS.NLM version prior to 6.00.	Refer to the instructions in the READUPDS.TXT file.

Important: To avoid NDS base schema conflicts, always upgrade the server holding the master replica of the [Root] partition first.

Setting Up a Lab

You should set up a lab should be set up to install, configure, and test NetWare 4.2 for your particular network environment. It will provide the hands-on experience important to developing an efficient migration strategy and implementation of NetWare 4.

The hardware and software should be representative of the existing network environment. Try to use similar network boards, LAN topology, and workstation operating systems. At least one CD-ROM player should be included for installing the NetWare 4 software on the initial server.

The lab environment should not affect the current operation of the existing network but should maintain a connection to the current network backbone. This will allow for migration and backward compatibility testing.

Using NetWare 4.2 Utilities

A full suite of utilities is provided for implementing NetWare 4. These utilities include NetWare Loadable Module (NLM) programs for the server and utilities for the workstation.

The NetWare 4 utilities support Windows, DOS, and NFS* UNIX environments.

Note: Because of differences between Novell Directory Services and the bindery, earlier versions of utilities and NLM programs do not always correspond to NetWare 4 utilities and NLM programs.

Server Utilities and NLM Programs

With NetWare 4, there are two types of utilities used at the server console:

- ◆ Command line utilities

Command line utilities are executed by typing the command as described in *Utilities Reference*.

- ◆ NetWare Loadable Module™ (NLM™) programs (typically, menu-based utilities)

NLM programs must be loaded from the server console prompt by typing the **LOAD** command followed by the NLM filename.

A list of all server utilities included with NetWare 4.2 and those that are new or updated since NetWare 3.1x is available in *Utilities Reference*.

NetWare Workstation Utilities

In NetWare 4, there are three types of utilities used at a workstation:

- ◆ DOS command line utilities

DOS command line utilities are executed by typing the command at a DOS prompt on a workstation or from within a login script or batch file as described in *Utilities Reference*.

- ◆ DOS menu-based utilities

DOS menu-based utilities are executed by typing the name of the utility at a DOS prompt on a workstation.

- ◆ Graphical utilities

Graphical utilities are run from within the Windows 3.1, Windows 95/98, or Windows NT environments.

A list of all server utilities included with NetWare 4 and those that are new or updated since NetWare 3.1x is available in *Utilities Reference*.

Analyzing Hardware and Hardware Driver Compatibility

Analyzing network hardware and hardware drivers allows you to determine what versions of drivers are being used and which operating systems these drives can support. Many manufacturers have developed specific hardware drivers for NetWare 4. Contact the hardware manufacturers for copies of the latest version of hardware drivers and information about product support.

Establishing a Pilot System

A pilot system provides a testing environment for evaluating and analyzing compatibility of network utilities and applications with NDS and bindery services. The following factors should be evaluated and analyzed:

- ◆ NetWare 4 installation options

Run the installation program to familiarize yourself with the various installation options. The process is simple; however, you might need to add new programs or additional licenses. Novell's online documentation is installed through the installation utility. Client diskettes are also created with the installation program.

- ◆ Initial Directory tree creation

The Directory tree foundation is created during the installation process. You should test your tree design by creating the actual tree structure that was developed during the design process. Ensure that you use the naming standards for your organization and create actual container and leaf objects that will exist in your network environment. Only the [Root] Organization object and first level of Organizational Unit objects are created with the installation utility. All other objects are created with NetWare Administrator or NETADMIN.

- ◆ Time synchronization configuration

The first server installed into the Directory tree is a Single Reference time server. The second and following servers installed into the tree are Secondary time servers. Once all the lab servers are installed, you should configure time synchronization for each lab server according to the established time synchronization strategy.

Use the SERVMAN server utility to change the time server setting if needed.

- ◆ Partition and replica creation

The [Root] partition is automatically created on the first NetWare 4 server during installation. Use NDS Manager or PARTMGR to create partitions and replicas in the tree. The number of servers in

your lab environment depends on how many partitions and replicas can be created.

- ◆ Application testing

Perform application testing to ensure compatibility between existing network environments and NetWare 4. Test client and server applications and the NetWare 4 and third-party applications that are used on your network.

See “Application Compatibility” on page 252 for a copy of a template you can use to perform your testing.

- ◆ Backup and restore procedures

The backup and restore process used on your network should be tested and evaluated for NetWare 4. Ensure that your backup utility is updated to perform both NetWare 4 data and NDS backup.

- ◆ Client installation or migration

Three options are available for installing or migrating client workstations to NetWare 4. These methods are: floppy diskette, CD-ROM, or across-the-wire. Identify which method you will use. Test any automation processes or configurations. The NET.CFG files for DOS and Windows workstations should be tested and optimized.

Summary

An efficient migration strategy for client workstations and servers will make the implementation process straightforward. The lab environment will give you the hands-on experience you need to effectively implement NetWare 4.

Evaluation

You should ensure that the following procedures have been accomplished before continuing:

1. Your migration plan for client workstations and servers maintains some identifiable procedures.
2. Responsibility for each implementation procedure has been assigned to a team member.

3. Strategies include configuration concerns for all network workstation types.
4. Individual server concerns are identified and a migration method is decided for each.
5. Adequate testing has been done to ensure seamless integration and compatibility.

Where to Go from Here

To	Go to
Configure and troubleshoot interoperability testing for NetWare 4	The Novell Application Notes at the Novell Web site.
Develop a schedule for implementing NetWare 4 on your network	Chapter 10, "Creating an Implementation Schedule," on page 197
Implement NetWare 4 on your network	Chapter 11, "Implementing NetWare 4 Services," on page 203

10 *Creating an Implementation Schedule*

If you have an existing NetWare[®] network, networks based on non-NetWare operating systems, or complex networks, you should create an implementation schedule to ensure that the implementation of NetWare 4 is as troublefree and efficient as possible.

This chapter describes how to create an implementation schedule for your implementation of the NetWare 4[™] operating system.

The following topics are discussed:

- ◆ “Identifying Tasks and Assignments” on page 198
- ◆ “Determining an Efficient Implementation Schedule” on page 199

If you have a new network with only one server, you do not need to create an implementation schedule. Continue to the next procedure. See Chapter 11, “Implementing NetWare 4 Services,” on page 203 for more information.

Introduction

A smooth implementation of NetWare 4 requires adequate planning to manage the various implementation phases. A well-planned implementation eliminates confusion, provides efficient execution of tasks, and communicates migration tasks.

Create a schedule to provide the necessary planning that your system needs. Some networks will develop a very short schedule and others may carry over a few months.

An efficient schedule gives you a timeline for the entire implementation process and determines the individual tasks that need to be completed.

Objectives

You should

1. Clearly identify the tasks that need to be accomplished.
2. Assign resources to execute the task.
3. Plan when the tasks will be started and finished.

Doing so will ensure that network client workstations and servers can continue to operate uninterrupted.

This enables you to organize your schedule by tasks and identify the general tasks needed to perform the implementation process.

Prerequisites

- You should have copies of the following planning documents:
 - ◆ Resource maps
 - ◆ Location maps
 - ◆ LAN and WAN topology maps
 - ◆ Directory tree design

Identifying Tasks and Assignments

Identifying tasks and assignments requires you to create a task list and then assign these tasks.

Creating a Task List

For each implementation task, your implementation schedule should include the task description, start date, target end date, the person assigned, and the percentage of completion.

Once the schedule is created, it can be used to set deadlines, measure the progress of tasks, review the work, and produce reports.

When creating a task list, consider the following elements of the implementation process:

- ◆ Client software migration or installation and configuration
- ◆ Server installation and configuration
- ◆ Directory tree design
- ◆ Access and security strategy
- ◆ Time synchronization strategy
- ◆ Partition and replication plan
- ◆ File and print services setup and configuration

Assigning Task Responsibilities

Once an implementation task list is created, then each task should be assigned to a person or team that is responsible for completion of the task. A person or team might be responsible for more than one task in the list.

The task owner should ensure that the person or team responsible for the task is familiar with the procedures and utilities necessary for completing their assignments.

Determining an Efficient Implementation Schedule

An efficient implementation schedule lays a foundation for improved communication and predictability by allowing you to visualize the implementation tasks in relationship to each other. This perspective allows you to reduce the amount of rework needed to manage the project effectively, and to deal assertively with issues that may arise.

Scheduling Tasks

You should identify a projected start and end date for each task. This allows the project team to better manage the sequence of interdependent tasks and workflows.

You must coordinate schedules with other groups in your organization. This will ensure that all groups are informed of the changes that will occur and allow them to train and prepare for each step of the implementation process.

Tracking Progress

Project tracking provides a clear indication of the project status and helps ensure a high level of communication about the progress of each task. Project tracking also allows the project team lead to address issues quickly and assertively before they become problems that may hinder progress.

Creating an Implementation Schedule Draft

Your team should draft a project schedule for implementing NetWare 4. The draft should include at least the following elements:

- ◆ A listing of each project task
- ◆ A description of each task
- ◆ The owner of each task
- ◆ A beginning and end date for each task
- ◆ A status indicator for each task

See “Implementation Schedule” on page 253 to view an example of an implementation schedule.

Summary

Creating an efficient implementation schedule requires you to complete the following tasks:

- ◆ Make a list of all the tasks necessary for implementing NetWare 4
- ◆ Match each task with the appropriate resource
- ◆ Determine appropriate timelines and milestones

Where to Go from Here

To	Go to
Implement NetWare 4 on your network	Chapter 11, "Implementing NetWare 4 Services," on page 203

This chapter describes the process used for implementing the NetWare[®] 4[™] operating system.

The following topics are discussed:

- ◆ “Completing General Tasks” on page 204
- ◆ “Implementing NDS on Various Network Types” on page 208

Introduction

Implementing the Novell[®] Directory Services[™] (NDS[™]) technology on your network can be as simple or complex as you want it to be. The flexibility of NDS allows you to install and run it on a single server or many servers.

Objectives

You should

1. Clearly understand the tasks that need to be accomplished
2. Know which resources are assigned to execute each task
3. Know when each task will be started and finished

By meeting these objectives, you will ensure that network client workstations and servers can continue to operate uninterrupted.

This will enable you to complete the implementation smoothly and on schedule.

Prerequisites

- You should have copies of the following planning documents:
 - ◆ Resource maps
 - ◆ Location maps
 - ◆ LAN and WAN topology maps
 - ◆ Directory tree design
 - ◆ Accessibility plan
 - ◆ Partition and replica worksheet
 - ◆ Implementation schedule

Completing General Tasks

To implement NDS on your network, you need to first complete the following general tasks:

1. Install the first server and set up the Directory tree.

When you install NetWare 4, the INSTALL utility automatically installs Novell Directory Services and prompts you to name the [Root] with the tree name. Then you can create and name an Organization object and up to three Organizational Unit objects.

You must then set the server context within the Directory tree. If you want to participate in the information superhighway, add a country code when setting the server context and a Country object will be created directly below the tree name or [Root] object.

The network hardware supports both file services and Novell Directory Services. If you add large numbers of leaf objects, such as users or print queues, to a single container, you might need to increase the amount of memory on that NetWare server to improve performance.

2. Migrate workstations

With the client software, you can upgrade network clients without modifying your existing server configurations. Your users will retain connectivity to all versions of NetWare. Once your client

workstations are migrated to the latest client software, your NetWare servers can be systematically migrated to NetWare 4 without interrupting user access to any server on the network.

With the Novell Client Software, you can install only the connectivity options needed on any workstation. For small networks, users can choose their optimal configuration to minimize memory use and maximize performance. For large network environments, you can establish a common configuration for multiple workstations.

3. Use NetWare Administrator or NETADMIN and PCONSOLE to complete the setup.

NetWare Administrator is a Windows-based utility, and NETADMIN and PCONSOLE are DOS-based utilities. To run these utilities, you must first install and set up a DOS or Windows* client workstation.

Then, you can set up the remaining Directory tree structure, create objects for all network resources you want available in the Directory database, and create Profile objects for maintenance purposes.

Leaf objects Place leaf objects in containers that provide the best access for the resources, groups, and users that use them.

For example, a NetWare Server object that stores a replica of each partition on the network can be placed in an Organization object for more efficient network management. Other servers and print queues can be placed in Organizational Units with the users or groups that use them.

Profile objects Create Profile objects that provide organization or department login scripts in the appropriate container objects for groups of users who need similar work environments but who are not located in the same container object.

Implementing objects this way allows for easy, centralized control at the top of the tree and local control of the lower levels. At each container level, a User object with supervisory rights has authority over the objects within that container object.

4. Add new servers to appropriate contexts.

To add a new server, first create the container object you want to install a new server into by using NetWare Administrator or NETADMIN.

5. Set the appropriate container and property rights.

Security features can be set up in NetWare Administrator or NETADMIN.

6. Configure time synchronization by specifying time synchronization parameters for each NetWare server.

The number and location of container objects, partitions, and replicas determine the type of time servers you should create for your network.

7. Make considerations for, and enable, bindery services by setting bindery contexts.

For security, optimum performance, and reliability, it is a good idea to group servers within container objects, depending on department or site. If, for example, your organization is spread over three cities, specify a site-specific container object as a bindery context for the following reasons:

- ◆ To provide local control over bindery services at each site

This allows the network supervisors to control local administration—updating local servers, adding or deleting users, installing new equipment, and performing other tasks that are often best handled on a local basis.

- ◆ To improve security

If, for example, network supervisors in three different cities have supervisor rights over the same container object (bindery context), each of them can assign rights that the other two would disagree with.

- ◆ To decrease traffic over WAN links

If, for example, users in London and Tokyo had their User objects in a bindery context served by a server in New York, every data transmission would take place over WAN links. This would likely result in decreased performance and create the potential for other problems.

To enable bindery services for objects within a container object, you need to set a bindery context to that container.

8. Optimize and manage Directory trees.

Use NDS Manager or PARTMGR to manage the Directory databases on your network.

Partitions Create most of your partitions at lower levels of the Directory tree. Workgroup boundaries generally determine the number of partitions required in a tree. You should partition your tree in relation to the use and physical locations of network resources. You should create partitions only if they provide better performance or fault tolerance to the network and tree.

Before performing any partition operation, ensure that the state of synchronization for all servers affected by the operation is stable. The following table provides recommendations for determining which partitions will be affected by what operation:

Partition Operation	Partitions Affected
Create, add, delete a partition	Target partitions
Change the replica type	Target partitions
Rebuild any partition replica	Target partitions
Join partitions	Parent and child partitions

Replicas Do not create too many replicas of the [Root] partition and other parent partitions. If you do, you force NDS to keep track of more child partition references than necessary. The appropriate number of replicas for any given partition depends on your environment.

If you create your Directory tree with the network user and resources in mind, you will find that the most efficient use of replicas—reducing WAN traffic while providing fault tolerance—means you should not need many replicas.

Implementing NDS on Various Network Types

The following discussions outline the recommended implementation of NDS features and functionality specific for single-site, multiple-site, and multiple-campus networks. You must decide which method or combination of methods best suits your organization's particular needs and requirements.

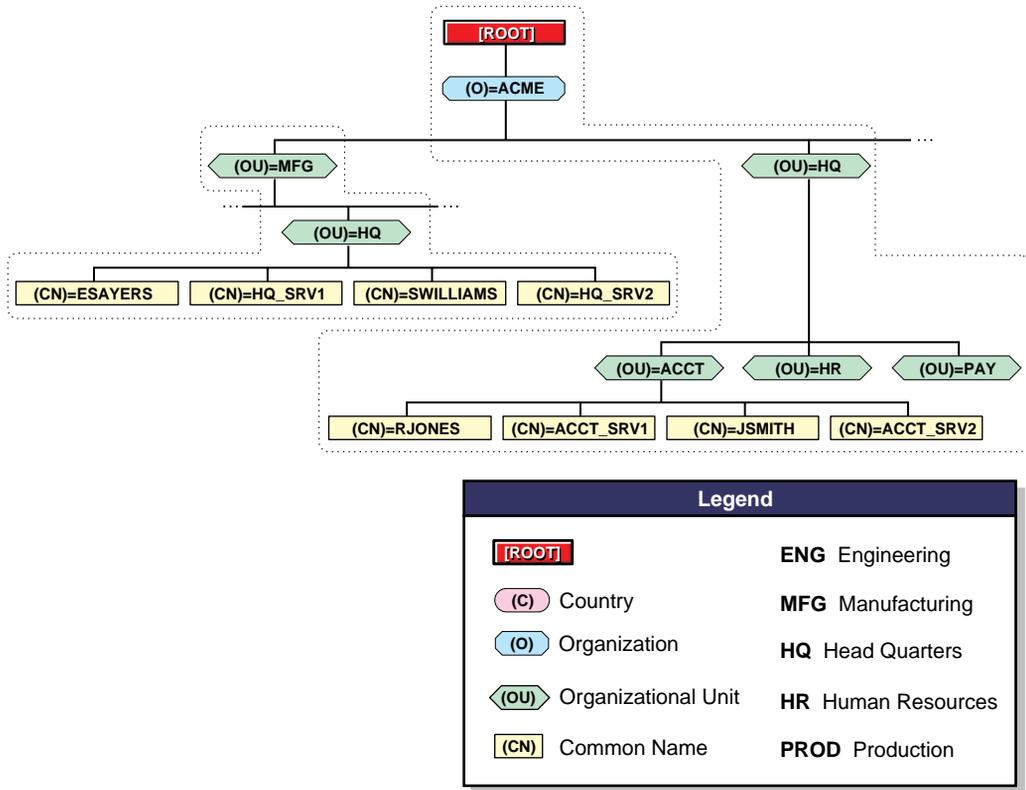
Single-Site Network

Implementing NDS on a single-site network is typically based on two possible models:

- ◆ The physical location of network resources
- ◆ The departmental structure of the organization

The following figure shows a Directory tree in which ACCT (Accounting), HR (Human Resources), and PAY (Payroll) represent departments, all under the Organization HQ (Headquarters).

Figure 11-1
 Example of a Single-Site Directory Tree



Directory Tree Structure

Single-site networks are commonly site-, workgroup-, and department-oriented in structure. They are easily managed by a system-wide administrative group with central management at the organizational and departmental levels.

The Directory tree begins with a single Organization object with few or no Organizational Unit objects below. If Organizational Units exist, they are based on functional groups, projects, and departments within a single site.

Resources are usually shared by all network users and groups.

Time Services

Although a single-site business might be restricted to a single- or multiple-segment LAN, time services is still important.

A Single Reference time server is usually adequate for LAN-based networks. The Single Reference time server is monitored and periodically adjusted for time by the network supervisors.

All other servers in the network are designated as Secondary time servers.

Partitions

Workgroup boundaries generally determine the number of partitions required in a tree. You should partition your tree in relation to the use and physical location of network resources. You should create partitions only if they provide better performance or fault tolerance to the network and tree. Single-site networks may not require partitioning.

If you think it is necessary, create a small number of partitions at the top levels of the tree.

Replicas

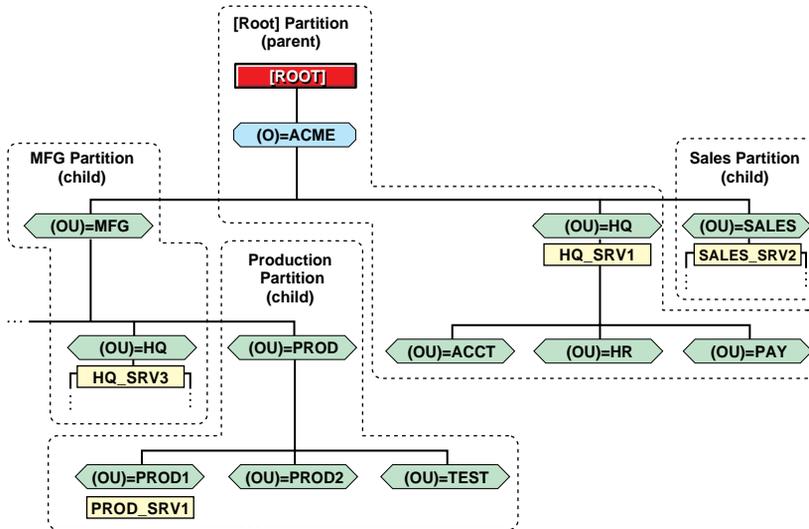
Each server on the network should contain all the resources needed for the users that it services. You should copy two to three replicas of each partition somewhere on the network to provide fault tolerance.

Multiple-Site Network

Implementing NDS on a multiple site network is typically based on your business' organizational chart with some geographic considerations for branch offices.

The following figure shows an example of a common Directory tree for a multiple-site network.

Figure 11-2
Example of Multiple-Site Directory Tree



		Partitions				
		[Root]	PROD	Sales	MFG	
Servers	1	PROD_SRV1	R/W	MR	SR	SR
	2	SALES_SRV2	R/W	SR	MR	R/W
	1	HQ_SRV3	R/W	SR	R/W	MR
	1	HQ_SRV1	MR	R/W	R/W	R/W

Legend		
[ROOT]	MR Master replica	S Single Reference time server
(C) Country	R/W Read/Write replica	R Reference time server
(O) Organization	RO Read-Only replica	1 Primary time server
(OU) Organizational Unit	SR Subordinate Reference replica	2 Secondary server
(CN) Common Name		

Directory Tree Structure

Multiple-site networks are commonly workgroup and department oriented in structure. They are typically managed by a central, system-wide administrative group and department network supervisors.

The Directory tree begins with a general Organization object with multiple Organizational Unit objects below. Organizational Units are based on functional groups, projects, departments, etc.

In the Organization object and high-level Organizational Units are enterprise resources that are managed centrally. For example:

- ◆ Servers that function as SAA* or TCP/IP gateways or as a NACS™ system
- ◆ User objects for network supervisors
- ◆ Profile objects that create an environment for specific users and groups

Create User objects for centralized supervisors and Organizational Unit object supervisors within their respective container objects. The Organizational Unit object-level supervisors are often department network supervisors.

Centralized supervisors are responsible for general network management and overall support for the Directory tree. Organizational Unit-level supervisors are responsible for day-to-day tasks, such as User object and resource management and local server backup.

Centralized management helps facilitate the implementation of network-wide standards. You should create and distribute a standards document for the entire network before implementing NDS.

Time Services

Because many multiple-site networks maintain some level of WAN connectivity, time services support is an important consideration.

A Single Reference time server is usually inadequate for networks that have WAN connections. You should use a group of Primary time servers as the basis for network time services.

Determine which servers within your organization provide system-wide services, such as directories or applications accessed by multiple departments or the entire organization.

Choose a limited number from the group of servers you identified to be installed as Primary time servers. Limiting the number of Primary time servers to a few minimizes the network traffic used when the time servers vote on the current time. Typically, you should have one or two Primary time servers at each location on the network.

Set up remaining servers as Secondary time servers.

Partitions

Partitioning multiple-site networks should follow the structure of your Organizational Unit objects. You might want to create a partition for each high-level Organizational Unit in the tree.

This allows each partition to contain all of the resource objects that a particular department needs to access. Place the [Root] and Organization objects in the same partition.

Replicas

Create replicas to ensure adequate redundancy of critical partitions. Determine which servers within your organization provide system-wide services, such as applications accessed by multiple departments or the entire organization.

Place replicas of the partitions that include these critical servers on other servers in different locations on the network. This allows all users to authenticate to an enterprise resource without increasing network traffic.

For servers that provide local services, place replicas of the partitions that include them on other local servers.

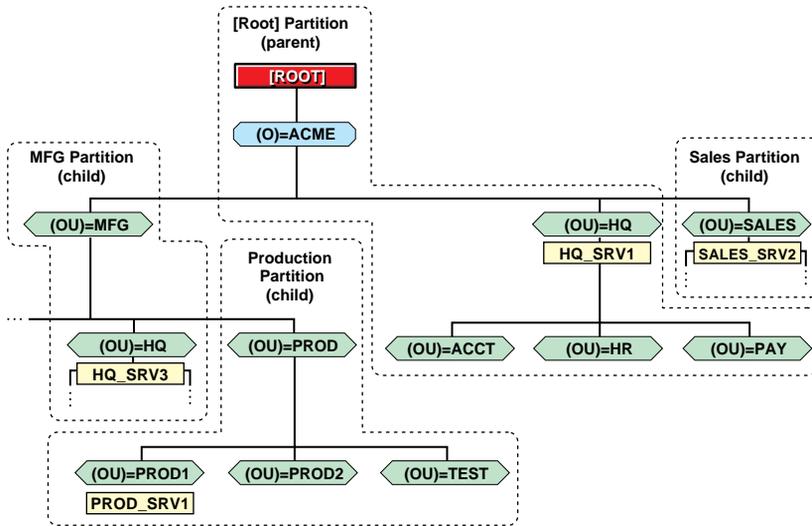
If only one server exists at a location, place a replica of the partition that includes the server on a server in a different location. Provide additional replicas if possible.

Multiple-Campus Network

Multiple-campus networks are enterprise focused, linking large, organizational networks with many other equal- or smaller-sized networks. They require flexibility, advanced security, and centralized management of distant resources as well as local supervision.

The following figure shows an example of a Directory tree for a multiple-campus network.

Figure 11-3
Example of Multiple-Campus Directory Tree



		Partitions				
		[Root]	PROD	Sales	MFG	
Servers	1	PROD_SRV1	R/W	MR	SR	SR
	2	SALES_SRV2	R/W	SR	MR	R/W
	1	HQ_SRV3	R/W	SR	R/W	MR
	1	HQ_SRV1	MR	R/W	R/W	R/W

Legend		
[ROOT]	MR Master replica	S Single Reference time server
(C) Country	R/W Read/Write replica	R Reference time server
(O) Organization	RO Read-Only replica	1 Primary time server
<OU> Organizational Unit	SR Subordinate Reference replica	2 Secondary server
(CN) Common Name		

Directory Tree Structure

The Directory tree begins with a general Organization object with multiple Organizational Unit objects below. Organizational Units are based on functional groups, projects, and departments and also on site locations such as cities or countries.

Multiple-campus networks typically require both system-wide administrative groups with central management at the organizational and departmental levels and site-based administrative groups that manage local resources and objects.

Multiple-campus networks typically have a number of high-level divisions within the organization that form the top level of Organizational Units. Most of these divisions are divided into sub-departments which form a second level of Organizational Units. A third level of Organizational Units might consist of locations or functional groups.

Organizational and Departmental Containers

Organization and Organizational Unit objects contain enterprise resources that are managed centrally. For example:

- ◆ Servers that function as SAA or TCP/IP gateways or as a NACS system
- ◆ User objects for network supervisors
- ◆ Profile objects that create an environment for specific users and groups

Centralized Management

As organizations grow, it is necessary to maintain the workgroup and departmental structure of an organization while sufficiently increasing the centralized administration.

You should create User objects for centralized supervisors and Organizational Unit-level supervisors within their respective container objects.

Centralized supervisors are responsible for general network management and overall support for the Directory tree. Organizational Unit-level supervisors are responsible for day-to-day tasks, such as User object and resource management and local file server backup.

Centralized management also helps facilitate the implementation of network-wide standards. You should create and distribute a standards document for the entire network before implementing NDS.

Time Services

Because most multiple-campus networks maintain high levels of WAN connectivity, which span time zones and international datelines, time services support requires careful planning.

It is critical to have a constant reference of time in order for NDS synchronization to take place. Time is also important to the proper execution of certain events and features, such as network backups and time-based security.

You should use one Reference time server and a group of Primary time servers as the basis for network time services in a time provider group. This ensures that a proper and accurate time reference is available at all times.

Determine which servers within your organization provide system-wide services, such as directories or applications accessed by the entire organization. From the servers you identify, select one to function as the Reference time server and set up the others as Primary time servers.

Each geographically distinct site should have at least one Primary time server.

All other NetWare servers in the network should be set up as Secondary time servers.

The Reference time server should be adjusted periodically by an outside time source.

Partitions

Partitioning of multiple-sized networks should follow a multitiered partition plan.

Each division-level Organizational Unit has its own partition representing that container and its objects. Each lower-level Organizational Unit is the root for a partition that includes itself and all the other container and leaf objects beneath it in that branch of the tree.

The [Root] and Organization objects should form one partition. This partitioning structure ensures that all of the critical access points in the tree are available and can be replicated for redundancy.

Replicas

Create replicas to ensure adequate redundancy of critical partitions. Determine which servers in your organization provide system-wide services, such as applications accessed by multiple departments or the entire organization.

Place replicas of the partitions that include these critical servers on other servers in different locations on the network. This allows all users to authenticate to an enterprise resource without increasing network traffic.

For servers that provide local services, place replicas of the partitions that include them on other local servers.

If only one server exists at a location, place a replica of the partition that includes the server on a server in a different location. Provide additional replicas if possible.

For added security and fault tolerance, place a read/write replica of each partition on a server at the Organization object level of each Directory tree. This enables the central network management staff to maintain a complete Directory database in one location.

Make sure that every partition has a sufficient number of replicas available on the network, including replicas on appropriate distant servers, to ensure fault tolerance and to decrease WAN link traffic.

Most replicas should be located on servers in the main corporate network, except for other locations that have multiple servers. In these cases, replicas of the appropriate partitions are located on all of these servers.

Summary

Implementing NetWare 4 requires you to complete the following tasks:

1. Finalize and use any planning documents you have created to make a list of the Directory objects you will install.
2. Sort Directory objects by location.
3. Sort objects into a logical hierarchy.
4. Install the first server and set up the Directory tree.
5. Use NetWare Administrator or NETADMIN and PCONSOLE to complete the setup.
6. Add new servers to appropriate contexts.
7. Set the appropriate container and property rights.
8. Configure time synchronization by specifying time synchronization parameters for each NetWare server process.
9. Make considerations for, and enable, bindery services by setting bindery contexts.
10. Optimize and manage Directory trees.

A *NDS Object Classes and Properties*

Overview

This appendix explains the object classes and properties available in the Novell® Directory Services™ architecture.

Note: Some Novell documents use the term *attributes* and *properties* interchangeably.

The following topics are discussed:

- ◆ “NDS Object Classes and Their Functions” on page 222
- ◆ “NDS Object Classes and Their Properties” on page 225

NDS Object Classes and Their Functions

This section lists the most common NDS object classes, explains what each is used for, and indicates where that type of object can be contained.

Table A-1
Object Class, Function, and Possible Containment

Object Class	Function	Possible Container
Alias	Redirects the path of the Directory tree branch or leaf object to another location for more convenient access	Depends on the containment rules mandated by the object class to which the Alias object points
Audit File	Defines container and volume audit trails	Organization Organizational Unit
Bindery Object	Represents an object upgraded from a bindery-based server that cannot be mapped to a Directory object	Organization Organizational Unit
Bindery Queue	Represents an object that has been created by bindery services to emulate user-defined Queue objects	None
CommExec	Represents a class used in NetWare MHS services	None
Computer	Represents network computers that are not file or print servers (such as gateways, routers, and sometimes workstations)	Organization Organizational Unit
Country	Defines an additional level of organization in the Directory tree	Root level
Directory Map	Represent the physical name of the file system directory path	Organization Organizational Unit
Device	Represent physical units that can communicate, such as a modem, printer, etc.	Organization Organizational Unit

Object Class	Function	Possible Container
External Entity	Used by services (such as messaging) to store information about entities (such as e-mail users) outside of the Directory	Organization Organizational Unit
Group	Defines an unordered list of users that comprise a group for the purpose of assigning access rights	Organization Organizational Unit
List	Defines an unordered set of names without implying security equivalence	Organization Organizational Unit
Locality	Defines geographical locations in the Directory tree	Country Locality Organization Organizational Unit
Message Routing Group	Represents a group of messaging servers with direct connectivity for transferring messages between any two of them	Organization Organizational Unit
Messaging Server	Represents a server that picks up messages submitted by messaging applications or transferred from another messaging server	Organization Organizational Unit
NetWare Server	Represents a server that provides file and other services	Organization Organizational Unit
Organization	Defines an organization in a network	Country, Root, or Locality level
Organizational Role	Defines a position or role in an organization for the purpose of assigning access rights	Organization Organizational Unit
Organizational Unit	Defines a subdivision in an organization to contain objects	Organization Organizational Unit
Print Server	Represents a network print server	Organization Organizational Unit
Printer	Represents a physical printing device on network	Organization Organizational Unit
Profile	Specifies a login script used by several users	Organization Organizational Unit

Object Class	Function	Possible Container
Queue	Represents a batch processing queue for printing on the network	Organization Organizational Unit
Unknown	Represents any object created by the server to restore an object whose base class is no longer defined by the schema	None
User	Represents a user on the network	Organization Organizational Unit
Volume	Represents a physical volume in a NetWare server	Organization Organizational Unit

NDS Object Classes and Their Properties

This section lists the most common Directory object classes and the properties associated with each.

Table A-2

Object Class and Properties

AFP Server	Object Class (Inherited from Top)	CN (Common Name) (Inherited from Server)
Unique to Class		
Serial Number	ACL	Account Balance
Supported Connections	Authority Revocation	Allow Unlimited Credit
	Backlink	Descriptions
	Bindery Property	Full Name
	CA Private Key	Host Device
	CA Public Key	L (Locality Name)
	Certificate Revocation	Minimum Account Balance
	Certificate Validity Interval	Network Address
	Cross Certificate Pair	O (Organization Name)
	Equivalent To Me	OU (Organizational Unit Name)
	Last Referenced Time	Private Key
	Obituary	Public Key
	Reference	Resource
	Revision	Security Equals
		Security Flags
		See Also
		Status
		User
		Version

Alias	Object Class	
Unique to Class	ACL	
Aliased Object Name	Authority Revocation	
	Backlink	
	Bindery Property	
	CA Private Key	
	CA Public Key	
	Certificate Revocation	
	Certificate Validity Interval	
	Cross Certificate Pair	
	Equivalent To Me	
	Last Referenced Time	
	Obituary	
	Reference	
	Revision	
Audit File	Object Class (Inherited from Top)	Audit Policy
	ACL	Audit Type
	Audit A Encryption Key	Back Link
	Audit B Encryption Key	Description
	Audit Contents	L (Locality Name)
	Audit Current Encryption Key	
	Audit Link List	
	Audit Path	
Bindery Object	Object Class (Inherited from Top)	Certificate Revocation
Unique to Class	ACL	Certificate Validity Interval
Bindery Object Restriction	Authority Revocation	Cross Certificate Pair
Bindery Type	Backlink	Equivalent To Me
CN (Common Name)	Bindery Property	Last Referenced Time
	CA Private Key	Obituary
	CA Public Key	Reference
		Revision

Bindery Queue	Object Class (Inherited from Top)	Common Name (Inherited from Resource)
Unique to Class		
Bindery Type	ACL Authority Revocation Backlink Bindery Property CA Private Key CA Public Key Certificate Revocation Certificate Validity Interval Cross Certificate Pair Equivalent To Me Last Referenced Time Obituary Reference Revision	Description Host Resource Name L (Locality Name) O (Organization Name) OU (Organizational Unit Name) See Also Queue Directory (Inherited from Queue) Device Host Server Network Address Operator Server User Volume
CommExec	Object Class (Inherited from Top)	CN (Common Name) (Inherited from Server)
Unique to Class		
Network Address Restriction	ACL Authority Revocation Backlink Bindery Property CA Private Key CA Public Key Certificate Revocation Certificate Validity Interval Cross Certificate Pair Equivalent To Me Last Referenced Time Obituary Reference Revision	Account Balance Allow Unlimited Credit Descriptions Full Name Host Device L (Locality Name) Minimum Account Balance Network Address O (Organization Name) OU (Organizational Unit Name) Private Key Public Key Resource Security Equals Security Flags See Also Status User Version

Computer Unique to Class Operator Server Status	Object Class (Inherited from Top) ACL Authority Revocation Backlink Bindery Property CA Private Key CA Public Key Certificate Revocation Certificate Validity Interval Cross Certificate Pair Equivalent To Me Last Referenced Time Obituary Reference Revision	CN (Common Name) (Inherited from Device) Descriptions L (Locality Name) Network Address O (Organization Name) OU (Organizational Unit Name) Owner See Also Status Serial Number
Country Unique to Class C (Country Name) Description	Object Class (Inherited from Top) ACL Authority Revocation Backlink Bindery Property CA Private Key CA Public Key	Certificate Revocation Certificate Validity Interval Cross Certificate Pair Equivalent To Me Last Referenced Time Obituary Reference Revision
Directory Map Unique to Class Host Server Path	Object Class (Inherited from Top) ACL Authority Revocation Backlink Bindery Property CA Private Key CA Public Key Certificate Revocation Certificate Validity Interval Cross Certificate Pair Equivalent To Me Last Referenced Time Obituary Reference Revision	CN (Common Name) (Inherited from Resource) Descriptions Host Resource Name L (Locality Name) O (Organization Name) OU (Organizational Unit Name) Owner See Also

External Entity	OU (Organizational Unit Name)	Object Class (Inherited from Top)
Unique to Class	Physical Delivery Office Name	ACL
CN (Common Name)	Postal Address	Authority Revocation
Description	Postal Code	Backlink
E-Mail Address	Postal Office Box	Bindery Property
External Entity	S (Street or Province Name)	CA Private Key
Facsimile Telephone Number	SA (Street Address)	CA Public Key
L (Locality Name)	See Also	Certificate Revocation
Mailbox ID	Title	Certificate Validity Interval
Mailbox Location		Cross Certificate Pair
		Equivalent To Me
		Last Referenced Time
		Obituary
		Reference
		Revision

Group	Object Class (Inherited from Top)
Unique to Class	ACL
CN (Common Name)	Authority Revocation
Description	Backlink
E-Mail Address	Bindery Property
Full Name	CA Private Key
GID (Group ID)	CA Public Key
L (Locality Name)	Certificate Revocation
Login Script	Certificate Validity Interval
Mailbox ID	Cross Certificate Pair
Mailbox Location	Equivalent To Me
Member	Last Referenced Time
O (Organization Name)	Obituary
OU (Organizational Unit Name)	Reference
Owner	Revision
Profile	
Profile Membership	
See Also	

List	Object Class (Inherited from Top)
Unique to Class	
CN (Common Name)	ACL
Description	Authority Revocation
Email Address	Backlink
Full Name	Bindery Property
Mailbox ID	CA Private Key
Mailbox Location	CA Public Key
Member	Certificate Revocation
O (Organization Name)	Certificate Validity Interval
OU (Organizational Unit Name)	Cross Certificate Pair
Owner	Equivalent To Me
See Also	Last Referenced Time
	Obituary
	Reference
	Revision

Locality	Object Class (Inherited from Top)
Unique to Class	
Description	ACL
L (Locality Name)	Authority Revocation
S (Street or Province Name)	Backlink
SA (Street Address)	Bindery Property
See Also	CA Private Key
	CA Public Key
	Certificate Revocation
	Certificate Validity Interval
	Cross Certificate Pair
	Equivalent To Me
	Last Referenced Time
	Obituary
	Reference
	Revision

<p>Message Routing Group</p> <p>Unique to Class</p> <p>None</p>	<p>Object Class (Inherited from Top)</p> <p>ACL</p> <p>Authority Revocation</p> <p>Backlink</p> <p>Bindery Property</p> <p>CA Private Key</p> <p>CA Public Key</p> <p>Certificate Revocation</p> <p>Certificate Validity Interval</p> <p>Cross Certificate Pair</p> <p>Equivalent To Me</p> <p>Last Referenced Time</p> <p>Obituary</p> <p>Reference</p> <p>Revision</p>	<p>CN (Common Name) (Inherited from Group)</p> <p>Description</p> <p>E-Mail Address</p> <p>Full Name</p> <p>GID (Group ID)</p> <p>L (Locality Name)</p> <p>Login Script</p> <p>Mailbox ID</p> <p>Mailbox Location</p> <p>Member</p> <p>O (Organization Name)</p> <p>OU (Organizational Unit Name)</p> <p>Owner</p> <p>Profile</p> <p>Profile Membership</p> <p>See Also</p>
<p>Messaging Server</p> <p>Unique to Class</p> <p>Message Routing Group</p> <p>Messaging Database</p> <p>Location</p> <p>Messaging Server Type</p> <p>Postmaster</p> <p>Supported Gateway</p> <p>Supported Services</p>	<p>Object Class (Inherited from Top)</p> <p>ACL</p> <p>Authority Revocation</p> <p>Backlink</p> <p>Bindery Property</p> <p>CA Private Key</p> <p>CA Public Key</p> <p>Certificate Revocation</p> <p>Certificate Validity Interval</p> <p>Cross Certificate Pair</p> <p>Equivalent To Me</p> <p>Last Referenced Time</p> <p>Obituary</p> <p>Reference</p> <p>Revision</p>	<p>CN (Common Name) (Inherited from Server)</p> <p>Account Balance</p> <p>Allow Unlimited Credit</p> <p>Description</p> <p>Full Name</p> <p>Host Device</p> <p>L (Locality Name)</p> <p>Minimum Account Balance</p> <p>Network Address</p> <p>O (Organization Name)</p> <p>OU (Organizational Unit Name)</p> <p>Private Key</p> <p>Public Key</p> <p>Resource</p> <p>Security Equals</p> <p>Security Flags</p> <p>See Also</p> <p>Status</p> <p>User</p> <p>Version</p>

NetWare Server	Object Class (Inherited from Top)	CN (Common Name) (Inherited from Server)
Unique to Class		
DS Revision	ACL	Account Balance
Message Server	Authority Revocation	Allow Unlimited Credit
Operator	Backlink	Description
Supported Services	Bindery Property	Full Name
	CA Private Key	Host Device
	CA Public Key	L (Locality Name)
	Certificate Revocation	Minimum Account Balance
	Certificate Validity Interval	Network Address
	Cross Certificate Pair	O (Organization Name)
	Equivalent To Me	OU (Organizational Unit Name)
	Last Referenced Time	Private Key
	Obituary	Public Key
	Reference	Resource
	Revision	Security Equals
		Security Flags
		See Also
		Status
		User
		Version

Organization	Login Intruder Limit	Object Class (Inherited from Top)
Unique to Class	Login Script	
	Mailbox ID	
O (Organization Name)	Mailbox Location	ACL
	NNS Domain	Authority Revocation
Description	Physical Delivery Office	Backlink
Detect Intruder	Postal Address	Bindery Property
EMail Address	Postal Code	CA Private Key
Facsimile Telephone	Postal Office Box	CA Public Key
Number	Print Job Configuration	Certificate Revocation
Intruder Attempt Reset	Printer Control	Certificate Validity Interval
Interval	S (State or Province Name)	Cross Certificate Pair
Intruder Lockout Reset	SA (Street Address)	Equivalent To Me
Interval	See Also	Last Referenced Time
L (Locality Name)	Telephone Number	Obituary
Lockout After Detection		Reference
		Revision

Organizational Role	Object Class (Inherited from Top)	
Unique to Class		
Description	ACL	
E-Mail Address	Authority Revocation	
Facsimile Telephone Number	Backlink	
L (Locality Name)	Bindery Property	
Mailbox ID	CA Private Key	
Mailbox Location	CA Public Key	
OU (Organizational Unit Name)	Certificate Revocation	
Physical Delivery Office Name	Certificate Validity Interval	
Postal Address	Cross Certificate Pair	
Postal Code	Equivalent To Me	
Postal Office Box	Last Referenced Time	
Role Occupant	Obituary	
S (State or Province Name)	Reference	
SA (Street Address)	Revision	
See Also		
Telephone Number		
Organization Unit	Login Intruder Limit	Object Class (Inherited from Top)
Unique to Class	Login Script	
OU (Organization Unit Name)	Mailbox ID	ACL
Description	Mailbox Location	Authority Revocation
Detect Intruder	NNS Domain	Backlink
E-Mail Address	Physical Delivery Office	Bindery Property
Facsimile Telephone Number	Postal Address	CA Private Key
Intruder Attempt Reset Interval	Postal Code	CA Public Key
Intruder Lockout Reset Interval	Postal Office Box	Certificate Revocation
L (Locality Name)	Print Job Configuration	Certificate Validity Interval
Lockout After Detection	Printer Control	Cross Certificate Pair
	S (State or Province Name)	Equivalent To Me
	SA (Street Address)	Last Referenced Time
	See Also	Obituary
	Telephone Number	Reference
		Revision

Print Server	Object Class (Inherited from Top)	CN (Common Name) (Inherited from Server)
Unique to Class		
Operator	ACL	Account Balance
Printer	Authority Revocation	Allow Unlimited Credit
SAP Name	Backlink	Description
	Bindery Property	Full Name
	CA Private Key	Host Device
	CA Public Key	L (Locality Name)
	Certificate Revocation	Minimum Account Balance
	Certificate Validity Interval	Network Address
	Cross Certificate Pair	O (Organization Name)
	Equivalent To Me	OU (Organizational Unit Name)
	Last Referenced Time	Private Key
	Obituary	Public Key
	Reference	Resource
	Revision	Security Equal To
		Security Flags
		See Also
		Status
		User
		Version

Printer	Object Class (Inherited from Top)	CN (Common Name) (Inherited from Device)
Unique to Class		
Cartridge	ACL	Description
Default Queue	Authority Revocation	L (Locality Name)
Host Device	Backlink	Network Address
Memory	Bindery Property	O (Organization Name)
Network Address	CA Private Key	OU (Organizational Unit Name)
Restriction	CA Public Key	Owner
Notify	Certificate Revocation	See Also
Operator	Certificate Validity Interval	Serial Number
Page Description	Cross Certificate Pair	
Language	Equivalent To Me	
Print Server	Last Referenced Time	
Printer Configuration	Obituary	
Queue	Reference	
Status	Revision	
Supported Typefaces		

Profile	Object Class (Inherited from Top)	
Unique to Class		
CN (Common Name)	ACL	
Login Script	Authority Revocation	
	Backlink	
Descriptions	Bindery Property	
Full Name	CA Private Key	
L (Locality Name)	CA Public Key	
O (Organization Name)	Certificate Revocation	
OU (Organizational Unit Name)	Certificate Validity Interval	
See Also	Cross Certificate Pair	
	Equivalent To Me	
	Last Referenced Time	
	Obituary	
	Reference	
	Revision	
Queue	Object Class (Inherited from Top)	CN (Common Name) (Inherited from Resource)
Unique to Class		
Device	ACL	Description
Host Server	Authority Revocation	Host Resource Name
Network Address	Backlink	L (Locality Name)
Operator	Bindery Property	O (Organization Name)
Server	CA Private Key	OU (Organizational Unit Name)
User	CA Public Key	Owner
Volume	Certificate Revocation	See Also
	Certificate Validity Interval	
	Cross Certificate Pair	
	Equivalent To Me	
	Last Referenced Time	
	Obituary	
	Reference	
	Revision	
Unknown	Object Class (Inherited from Top)	Certificate Revocation
Unique to Class		Certificate Validity Interval
None	ACL	Cross Certificate Pair
	Authority Revocation	Equivalent To Me
	Backlink	Last Referenced Time
	Bindery Property	Obituary
	CA Private Key	Reference
	CA Public Key	Revision

User	Password Unique Required	CN (Common Name) (Inherited from Person)
Unique to Class	Password Used	Surname
Account Balance	Print Job Configuration	Description
Allow Unlimited Credit	Printer Control	Full Name
Group Membership	Private Key	Generational Qualifier
Higher Privileges	Profile	Glven Name
Home Directory	Profile Membership	Initials
Language	Public Key	See Also
Last Login Time	Security Equal To	Telephone Number
Locked By Intruder	Security Flags	None (Inherited from Organizational Person)
Login Allowed Time Map	Server Holds	E-Mail Address
Login Disabled	Type Creator Map	Facsimile Telephone Number
Login Expiration Time	UID (User ID)	L (Locality Name)
Login Grace Limit	Object Class (Inherited from Top)	Mailbox ID
Login Grace Remaining	ACL	Mailbox Location
Login Intruder Address	Authority Revocation	OU (Organizational Unit Name)
Login Intruder Attempts	Backlink	Physical Delivery Office Name
Login Intruder Reset Time	Bindery Property	Postal Address
Login Maximum	CA Private Key	Postal Code
Simultaneous	CA Public Key	Postal Office Box
Login Script	Certificate Revocation	Role Occupant
Login Time	Certificate Validity Interval	S (State or Province Name)
Message Server	Cross Certificate Pair	SA (Street Address)
Minimum Account Balance	Equivalent To Me	Title
Network Address	Last Referenced Time	
Network Address	Obituary	
Restriction	Reference	
Password Allow Change	Revision	
Password Expiration		
Interval		
Password Expiration Time		
Password Minimum		
Length		
Password Required		

Volume	Object Class (Inherited from Top)	CN (Common Name) (Inherited from Resource)
Unique to Class		
Host Server	ACL	Description
Status	Authority Revocation	Host Resource Name
	Backlink	L (Locality Name)
	Bindery Property	O (Organization Name)
	CA Private Key	OU (Organizational Unit Name)
	CA Public Key	See Also
	Certificate Revocation	
	Certificate Validity Interval	
	Cross Certificate Pair	
	Equivalent To Me	
	Last Referenced Time	
	Obituary	
	Reference	
	Revision	

Overview

Directory leaf objects are objects that do not contain any other objects. These represent actual network entities such as users, servers, printers, and computers. You create leaf objects in a container object.

This appendix explains the leaf objects available in the Novell® Directory Services™ architecture.

The following topics are discussed:

- ◆ “Application Leaf Object” on page 240
- ◆ “Auditing Leaf Object” on page 240
- ◆ “Informational Leaf Object” on page 241
- ◆ “Messaging-Related Leaf Objects” on page 241
- ◆ “Miscellaneous Leaf Object” on page 243
- ◆ “NetWare Licensing Services Leaf Object” on page 244
- ◆ “Printer-Related Leaf Objects” on page 245
- ◆ “Server-Related Leaf Objects” on page 246
- ◆ “User-Related Leaf Objects” on page 247

Application Leaf Object

This section describes the leaf objects that support the NetWare® Application Manager, explains what it is used for, and indicates when to use it.

Table B-1

Application Leaf Object

Leaf Object	Function	Usage Situation
Application	<p>NetWare Application Manager allows network supervisors to manage network applications as Application objects in the Novell Directory tree.</p> <p>NetWare Application Manager displays icons for all available applications in a window, and lets the user select on an icon to launch an application. Users don't need to worry about drive mappings, paths, or rights.</p> <p>The network supervisor can manage applications at the container, Group, or User object level.</p>	<p>Application objects allow you to manage the network more efficiently, saving time and effort administering applications.</p> <p>Application objects simplify administrative tasks such as assigning rights, customizing login scripts, and supporting applications.</p>

Auditing Leaf Object

This section describes the leaf object for auditing network resources, explains what it is used for, and indicates when to use it.

Table B-2

Auditing Leaf Object

Leaf Object	Function	Usage Situation
Auditing File	<p>The Auditing File Object (AFO) is the Novell Directory Services data structure used to manage an audit trail's configuration and access rights.</p>	<p>An audit utility (such as AUDITCON) creates the AFO when you enable auditing. The server then checks for access rights each time a user attempts to access the audit trail.</p>

Informational Leaf Object

This section describes the leaf object that exists only to store information about network resources, explains what it is used for, and indicates when to use it.

Table B-3

Informational Leaf Object

Leaf Object	Function	Usage Situation
Computer	Represents a nonserver computer on the network, such as a workstation or a router.	<p>Use this object to store information about a nonserver computer, such as its network address, serial number, or person the computer is assigned to.</p> <p>This object has no effect on the operation of the network; it only stores information about the computer.</p>

Messaging-Related Leaf Objects

This section describes the leaf objects that are related to the NetWare Message Handling Service™ (MHS) system, explains what each is used for, and indicates when to use each.

These objects are created and controlled using the MHS utilities. MHS Services software is available on NetWare® .

Table B-4

Messaging-Related Leaf Objects

Leaf Object	Function	Usage Situation
Distribution List	Represents a list of mail recipients.	<p>Use this object to simplify sending mail to recipients.</p> <p>For example, you could create a Distribution List object called Recreation Committee. Anyone wanting to send a message to all the members in the Recreation Committee can simply address the message to Recreation Committee, rather than to each member.</p>

Leaf Object	Function	Usage Situation
External Entity	<p>Represents a non-native NDS object that is imported into NDS or registered in NDS.</p> <p>NetWare MHS Services use External Entity objects to represent users from non-NDS directories to provide an integrated address book for sending mail.</p> <p>MHS Services software is available on NetWare.</p>	<p>If your messaging environment contains non-MHS servers, such as SMTP hosts, SNADS nodes, or X.400 MTAs, you might choose to add users and lists at these servers to your Directory database as External Entity objects.</p> <p>Adding these objects to the database as External Entities adds them to the address books of your messaging applications. When addressing messages, local users can choose non-MHS users and lists from a directory list.</p>
Message Routing Group	<p>Represents a group of messaging servers that can transfer messages directly with each other.</p> <p>MHS Services software is available on NetWare.</p>	<p>Create a Message Routing Group when you have two or more messaging servers that need to communicate with each other.</p>
Messaging Server	<p>Represents a messaging server that resides on a NetWare server. A Messaging Server object is automatically created in the Directory tree when you install NetWare MHS Services on a NetWare server.</p> <p>MHS Services software is available on NetWare.</p>	<p>Create a Messaging Server (by installing NetWare MHS Services on a NetWare server) when you need a server to handle messaging between users and groups on the network.</p>

Miscellaneous Leaf Object

This section lists the remaining available leaf objects, explains what each is used for, and indicates when to use each.

Table B-5

Miscellaneous Leaf Objects

Leaf Object	Function	Usage Situation
Alias	<p>Points to another object in the Directory tree and makes it appear as if the object that it points to actually exists in the Directory tree where the Alias object is.</p> <p>Although an object appears both where it was actually created and where an Alias referring to it was created, only one copy of the object really exists.</p> <p>If you delete or rename an Alias, the Alias itself (not the object it is pointing to) is deleted or renamed.</p>	<p>Use this object to allow access to an object that is in another context.</p> <p>For example, you can use an Alias to represent a resource, such as a special printer, that most users in the tree need to access.</p> <p>Also, when you move or rename a container object in a Directory tree, you have the option of creating an Alias object in place of the moved or renamed object. If you select this option, NetWare Administrator automatically creates the Alias for you and assigns it the same name as the original object.</p> <p>Creating an Alias object in place of a moved or renamed container object allows users to continue logging into the network and to see the container object (and the objects it contains) in its original Directory location.</p>
Bindery Object	<p>Represents an object placed in the Directory tree by an upgrade or migration utility.</p>	<p>It is used by NDS only to provide backward compatibility with bindery-based utilities.</p>
Bindery Queue	<p>Represents a queue placed in the Directory tree by an upgrade or migration utility.</p>	<p>It is used by NDS only to provide backward compatibility with bindery-based utilities.</p>
Unknown	<p>Represents an NDS object that has been invalidated and cannot be identified as belonging to any of the other object classes.</p>	<p>Directory Services utilities rename objects that they do not recognize. Delete or re-create the correct object for the resource.</p>

NetWare Licensing Services Leaf Object

This section describes the leaf object for NetWare Licensing Services (NLS), explains what it is used for, and indicates when to use it.

Table B-6

NetWare Licensing Services Leaf Object

Leaf Object	Function	Usage Situation
LSP Server	<p>When you register an LSP (License Service Provider) with NDS, an LSP Server object is created, by default, in the same context as the NCP Server object on which it is loaded. The LSP Server object can be relocated to another context in the Directory.</p> <p>An LSP Server object appears only if you load the NLS (NetWare Licensing Services) NLM program with an <code>-r</code> option.</p>	<p>A client-specific component packages the request and submits it to the nearest connected LSP.</p> <p>If the client is not connected to an LSP, the client checks the Novell Directory database, searching up the Directory tree for an LSP Server object.</p>

Printer-Related Leaf Objects

This section lists the leaf objects that are related to NetWare print services, explains what each is used for, and indicates when to use each.

These objects are created and controlled using the NetWare print utilities.

Table B-7

Printer-Related Leaf Objects

Leaf Object	Function	Usage Situation
Print Queue	Represents a print queue on the network.	You must create a Print Queue object for every print queue on the network. This object cannot be created with NETADMIN. Use the NetWare Administrator or PCONSOLE.
Print Server	Represents a network print server.	You must create a Print Server object for every print server on the network. This object cannot be created with NETADMIN. Use NetWare Administrator or PCONSOLE.
Printer	Represents a physical printing device on the network.	You must create a Printer object for every printer on the network. This object cannot be created with NETADMIN. Use NetWare Administrator or PCONSOLE.

Server-Related Leaf Objects

This section lists the leaf objects that are related to NetWare servers and volumes, explains what each is used for, and indicates when to use each.

Table B-8

Server-Related Leaf Objects

Leaf Object	Function	Usage Situation
Directory Map	<p>Represents a particular directory in the file system. Directory Map objects can be especially useful in login scripts by pointing to directories that contain applications or other frequently used files.</p> <p>For example, if you have a directory that contains DOS 6.0, you will probably map a search drive to that directory in any login scripts you create. Later, if you upgrade to DOS 6.2 and rename the directory, you have to change the mapping in every login script where that search mapping appears.</p> <p>If you use a Directory Map object, you only change the information in that object, not in all login scripts.</p>	<p>Use this object to avoid making changes to many login scripts when the location of applications changes; instead, you change only the Directory Map object.</p>
NetWare Server	<p>Represents a server running NetWare.</p> <p>Store information about the server in the NetWare Server object's properties, such as the server's address, the physical location of the server, and what services it provides.</p> <p>In addition to storing information about the NetWare server, the NetWare Server object affects the network in that it is referred to by several other objects.</p>	<p>Use the NetWare Server object to tie the physical server to the Directory tree. Without this object, you cannot access file systems that are on that server's volumes.</p> <p>If you have a non-NetWare 4 server, you must create this object to be able to access those non-NetWare 4 volumes. When you create a NetWare Server object for a non-NetWare 4 server, the non-NetWare 4 server must be running.</p>

Leaf Object	Function	Usage Situation
Volume	<p>Represents a physical volume on the network.</p> <p>In the Volume object's properties, you can enter identification information, such as the Host server, volume location, etc. You can also set restrictions for use of the volume, such as space limits for users.</p>	<p>You can create a Volume object for every physical volume on the network. (During installation of NetWare 4 on a server, Volume objects are created for every physical volume on that server.)</p> <p>Use INSTALL to create volumes on NetWare 4 servers.</p> <p>When a Volume object is created, the server name and the volume name information is placed in the Volume object's properties.</p> <p>You can use the Volume object to display information about the directories and files on that volume.</p>

User-Related Leaf Objects

This section lists the leaf objects that are related to network users and groups, explains what each is used for, and indicates when to use each.

Table B-9

User-Related Leaf Objects

Leaf Object	Function	Usage Situation
Group	<p>Assigns a name to a list of User objects that can be located anywhere in the Directory tree.</p>	<p>Create a Group when you have many User objects that need the same trustee assignments. Rather than making many trustee assignments, you make just one trustee assignment to all the users who belong to the group by making the trustee assignment to the Group object itself.</p>

Leaf Object	Function	Usage Situation
Organizational Role	Defines a position or role within an organization.	<p>Create an Organizational Role object so that you can assign rights to a particular position rather than to the person who may occupy that position. The occupant may change frequently, but the responsibilities of that position do not.</p> <p>You can assign any user to be an occupant of the Organizational Role object because every occupant receives the same rights that you granted to the Organizational Role object.</p>
Profile	Contains a profile login script. When the Profile object is listed as a User object's property, the Profile object's login script is executed when that User object logs in. The Profile login script executes after the system login script and before the user login script.	Create a Profile object for a set of users who need to share common login script commands but who are not located in the same container in the Directory tree, or who are a subset of users in the same container.

Leaf Object	Function	Usage Situation
User	<p data-bbox="408 181 738 234">Represents a person who uses your network.</p> <p data-bbox="408 261 760 402">In the User object properties, you can set login restrictions, intruder detection limits, password and password restrictions, security equivalences, etc.</p>	<p data-bbox="821 181 1207 437">You must create a User object for every user who needs to log in to the network. When you create a User object, you can create a home directory for that user. When you create User objects, you can also choose to apply a user template to the User object that provides default property values.</p> <p data-bbox="821 463 1207 693">For users who have NetWare 4 workstations, you can create the User objects anywhere in the Directory tree, but the users must know their context in order to log in. You should create User objects in the container where the users will typically log in.</p> <p data-bbox="821 719 1207 1037">For users who have non-NetWare 4 workstations, you must create the User objects in the container where the bindery services context is set for the server that they need to log in to. (Bindery services is set by default for every NetWare 4 server that is installed.) Non-NetWare 4 users do not need to know their context because they log in to the server rather than to the Directory tree.</p>

Template Examples

This appendix provides template examples that can be used for designing, implementing, and maintaining your NetWare® 4™ network.

You should customize all these templates examples to fit your specific network environment.

The following templates examples are provided on the indicated pages.

- ◆ “Application Compatibility” on page 252
- ◆ “Implementation Schedule” on page 253
- ◆ “Name Standards” on page 254
- ◆ “NetWare 4 Server Worksheet” on page 255
- ◆ “Replica Placement Template” on page 257
- ◆ “Server Migration” on page 258
- ◆ “Workstation Configuration Worksheet” on page 259

Application Compatibility

The following template provides an example of an application compatibility template you could use for your network migration.

Figure C-1
Application Compatibility Template

SoftWare and Version	Date Scheduled	VLM Tested
Word Processors		
Spreadsheets		
Email		
Menu Systems		
Databases		
Print Servers Software and Hardware Devices (Network Direct Connected)		
Gateways		
Tape Backup Systems		

Implementation Schedule

The following template provides an example of an implementation schedule template.

Figure C-2
Implementation Schedule Template

Task Title and Description	Owner	Begin Date	Target End Date	Percent Complete
Complete hardware setup (servers, workstations, cabling, routers, bridges, switches, peripherals)				
Install online documentation				
Prepare existing servers for migration				
Install and configure first server (Name the Directory tree and setup time synchronization)				
Install or migrate client workstations				
Setup and configure administration utilities				
Setup the Directory tree structure				
Implement your partition strategy				
Install or migrate remaining servers (Place servers in the appropriate Directory tree containers and setup time synchronization)				
Place replicas on appropriate NetWare servers according to your replication strategy				
Configure time synchronization to fit your time synchronization strategy				
Create appropriate objects for network administration				
Assign container level object rights				

Name Standards

The following template provides an example of an Novell® Directory Services™ (NDS™) naming standards document.

Figure C-3
Name Standards Worksheet Template

Item	Standard	Examples	Rationale
User object common name (login name)			
User object last name			
User object telephone and fax			
User object location			
Directory tree			
Organization			
Organizational Units whose names are based on a major location			
Other OU names			
Common names of leaf objects other than users			
Special objects <ul style="list-style-type: none"> • Organizational Role • Profile • Directory Map 			
All			

NetWare 4 Server Worksheet

The following templates provide a two-part example of a server worksheet template.

Figure C-4
Server Worksheet Template

Server information									
Server name: _____					IPX internal network number: _____				
Memory (RAM): Base: _____		Extended: _____			Total: _____				
Server boot method: Hard disk <input type="checkbox"/> Floppy diskette <input type="checkbox"/> 3.5" <input type="checkbox"/> 5.25"									
Network boards (Fill in columns that apply to each network board.)									
Name	LAN driver	I/O port	Memory address	Interrupt (IRQ)	DMA channel	Node address	Slot number	IPX external network #	Frame type
Other boards (Internal or external disk controllers, serial controllers, SCSI controllers, video adapters, etc.)									
Name	Driver (if applicable)	I/O port	Memory address	Interrupt (IRQ)	DMA channel	SCSI address	Other info		
Disks									
Drive Make/Model	Size	Mirrored with #	Volume segments						
1.									
2.									
3.									
4.									
5.									
6.									
7.									
8.									

Figure C-5

Server Worksheet Template

Directory tree information							
Server name: _____							
Directory context: _____				Tree name: _____			
Time Configuration							
Time server type: Single Reference <input type="checkbox"/> Reference <input type="checkbox"/> Primary <input type="checkbox"/> Secondary <input type="checkbox"/>							
Time zone: _____ Offset from UTC _____ Daylight Savings: Yes <input type="checkbox"/> No <input type="checkbox"/>							
Volumes							
Volume name	File compression		Block suballocation		Data migration		Name space
	ON	OFF	ON	OFF	ON	OFF	
NetWare Loadable Modules (NLM) Configuration							
NLM name	Configuration						

Workstation Configuration Worksheet

The following template provides an example of a workstation. configuration template.

Figure C-8
Workstation Worksheet Template

Workstation information									
Boot method: Hard disk <input type="checkbox"/> Floppy disk <input type="checkbox"/> Remote <input type="checkbox"/> Path: _____ Disk Size: _____ Memory (RAM): Base: _____ Extended: _____ Total: _____									
Directory tree information									
Directory context: _____					Tree name: _____				
Operating System									
DOS <input type="checkbox"/> _____			Macintosh <input type="checkbox"/> _____			Miscrosoft <input type="checkbox"/> _____			
OS/2 <input type="checkbox"/> _____			UNIX NFS <input type="checkbox"/> _____			Windows			
Time Configuration									
Time zone: _____			Offset from UTC _____			Sync with server: Yes <input type="checkbox"/> No <input type="checkbox"/>			
Network boards (Fill in columns that apply to each network board.)									
Name	LAN driver	I/O port	Memory address	Interrupt (IRQ)	DMA channel	Node address	Slot number	IPX external network #	Frame type
Other boards (Internal or external disk controllers, serial controllers, SCSI controllers, video adapters, etc.)									
Name	Driver (if applicable)	I/O port	Memory address	Interrupt (IRQ)	DMA channel	SCSI address	Other info		
Disks									
Drive Make/Model			Size	Volume name					
1.									
2.									
3.									

appendix **D**

Supplemental Information

The following publications provide supplemental information specifically related to NetWare® 4™ services and software.

Resources

Following are some additional information resources on NetWare 4, Novell® Directory Services™ (NDS™), SMS™, and related topics.

The Novell Application Notes (AppNotes) are an excellent source for information on topics related to NetWare. See developer.novell.com/research/appnotes.htm at the Novell Web site for a listing of all the notes published since 1990.

You can also check for the most recent versions of NetWare third-party books at your favorite bookstore.

The following FaxBack documents may also be helpful:

- ◆ *Storage Management Systems (SMS) Model* Doc. No. 1006
- ◆ *Backup and Restore of NDS* Doc. No. 1007

Online Help

- ◆ **Context-sensitive help**

If you are using a NetWare menu utility and want more information about how to complete a task, press <F1>.

If you are unsure how to use a command, type the command name and add the `/?` option for help. For example, for help with the `RIGHTS` command, type `RIGHTS /?`.

- ◆ **Online Windows help**

The Microsoft Windows help viewer allows you to read NetWare help developed for the Windows environment. To access the NetWare help screens within Windows, press F1 or the Help button.

- ◆ **Online documentation**

The viewer allows you to read NetWare documentation from your DOS, Windows, or UNIX workstation.

All NetWare 4™ documentation is available on the *NetWare Online Documentation* CD-ROM.

Additional Help Resources

- ◆ **Customer service**

You can contact your Novell Authorized Reseller^{CLM} representative for technical assistance.

Most Novell Authorized Resellers have Certified NetWare EngineerSM representatives on their staffs ready to assist users with their networking problems.

- ◆ **Novell Authorized Service Center (NASC) locations**

NASCSM facilities are local support providers authorized and supported by Novell. They provide both telephone and on-site assistance, and should be your first source for technical support.

For the Novell Authorized Service CenterSM nearest you, in the U.S. and Canada call 1-800-338-NASC.

- ◆ **Hardware documentation**

Many network problems occur because of malfunctioning hardware.

If you can isolate a problem to a certain computer component or cable segment, check the manuals that came with the hardware involved.

◆ **ManageWise services**

ManageWise® services help you manage the cabling system, computers, software, and other components of the network.

For more information about using NMS on your network, contact your Novell Authorized Reseller.

◆ **Other Novell publications**

Novell Application Notes and the Novell Research Reports™ publications cover technical aspects of NetWare-based system design, implementation, and management. *Application Notes* is a collection of technical articles published monthly. *Research Reports* is published as the research becomes available.

To purchase subscriptions and back issues of these publications from within the United States or Canada, call the Novell Research Order Desk at 1-800-377-4136. From other locations, call 303-297-2725.

◆ **Third-party books and periodicals**

Books on NetWare, including books published by Novell Press™ publishing, are available at most bookstores.

In addition, numerous networking periodicals give advice on configuring, managing, and troubleshooting your network.

◆ **NetWire forum on the CompuServe* bulletin board**

A fairly inexpensive way to get up-to-date advice and patches is through NetWire® on the CompuServe* bulletin board.

To open a CompuServe account, call one of the following numbers and ask for Representative 200:

- ◆ In the United States or Canada: 1-800-524-3388
- ◆ In the United Kingdom: 0800-289-378
- ◆ In Germany: 0130-37-32
- ◆ In other European countries: 44-272-255-111
- ◆ In locations other than the United States, Canada, or Europe, use the appropriate country code for the U.S. and call 614-457-0802. Ask for Representative 200. This identifies you as a Novell customer.

◆ **Technical Support Database and NetWire forum on the Internet**

The Novell internet sites support access through FTP, Gopher, and World Wide Web (WWW) systems. Over 9,000 documents exist on the WWW system for providing technical hints and information.

To access the Novell Internet sites, log in as ANONYMOUS and use your e-mail address as your password.

Contact one of the following site addresses:

- ◆ In the United States: [ftp.novell.com](ftp://ftp.novell.com)
- ◆ In Germany: [ftp.novell.de](ftp://ftp.novell.de)

See public areas in these sites for possible listings of other sites' addresses.

To access the Novell Internet WWW sites, contact one of the following site addresses:

- ◆ In the United States: www.novell.com and education.novell.com.
- ◆ In Germany: www.novell.de

See information areas in these sites for possible listings of other sites' addresses.

◆ **FaxBack Service**

Novell provides a FaxBack* service for obtaining additional product information to help with support needs.

To access the FaxBack service, complete the following steps.

- ◆ Access the FaxBack Service on the Internet at: netwire.novell.com/home/client/misc/catalog.htm.
- ◆ Within the continental United States
 1. Dial 1-800-NETWARE (1-800-638-9273).
 2. Press #1 (the Presale Product Information and Upgrade Information option).
 3. Press #2 (the Receive Product Information option).
 4. Press #1 (the Receive Product Information via Fax option).

- ◆ Outside the continental United States
Dial 1-801-861-2772. You are connected directly to the FaxBack service.
- ◆ Follow the directions provided on the phone. You are prompted to enter a document number and then a fax number to send the document to.
- ◆ **Network Support Encyclopedia Professional Volume™ (NSE Pro) package**

This encyclopedia gives customers access to regularly updated information on products and services—plus patches, fixes, and more—from Novell and other vendors.

The NSE Pro™ package is distributed on CD-ROM on a subscription basis. Updates are sent out several times each year. More information is available on NetWire or from your Novell Authorized Reseller.
- ◆ **Novell Consulting Services (NCS) Toolkit**

This toolkit is a collection of documents and utilities developed and collected by Novell's Consulting Services group for providing solutions to enterprise networks. It is packaged as a single CD-ROM. For information on obtaining a copy, contact Novell Consulting Services for details.
- ◆ **Troubleshooting hardware and software**

Specialized hardware and software packages, such as the Novell LANalyzer® software, are available to help you isolate network problems.

Trademarks

Novell Trademarks

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

Advanced NetWare is a trademark of Novell, Inc.

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

AppNotes is a trademark of Novell, Inc.

AppTester is a trademark of Novell, Inc. in the United States.

BrainShare is a registered service mark of Novell, Inc. in the United States and other countries.

C-Worthy is a trademark of Novell, Inc.

C3PO is a trademark of Novell, Inc.

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

Certified NetWare Administrator in Japanese and CNA-J are service marks of Novell, Inc.

Certified NetWare Engineer in Japanese and CNE-J are service marks of Novell, Inc.

Certified NetWare Instructor in Japanese and CNI-J are service marks of Novell, Inc.

Certified Novell Administrator and CNA are service marks of Novell, Inc.

Certified Novell Engineer and CNE are service marks of Novell, Inc.

Certified Novell Salesperson is a trademark of Novell, Inc.

Client 32 is a trademark of Novell, Inc.

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

Connectware is a trademark of Novell, Inc.

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

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

Custom 3rd-Party Object and C3PO are trademarks of Novell, Inc.

DeveloperNet is a trademark of Novell, Inc.

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

ElectroText is a trademark of Novell, Inc.

Enterprise Certified Novell Engineer and ECNE are service marks of Novell, Inc.

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

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

EXOS is a trademark of Novell, Inc.

Global MHS is a trademark of Novell, Inc.

Global Network Operations Center and GNOC are service marks of Novell, Inc.

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

Graphics Environment Manager and GEM are registered trademarks of Novell, Inc. in the United States and other countries.

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

GroupWise 5 is a trademark of Novell, Inc.

GroupWise XTD is a trademark of Novell, Inc.

Hardware Specific Module and HSM are trademarks of Novell, Inc.

Hot Fix is a trademark of Novell, Inc.

InForms is a trademark of Novell, Inc.

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

Internetwork Packet Exchange and IPX are trademarks of Novell, Inc.

IPX/SPX is a trademark of Novell, Inc.

IPXODI is a trademark of Novell, Inc.

IPXWAN is a trademark of Novell, Inc.

LAN WorkGroup is a trademark of Novell, Inc.

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

LAN WorkShop is a trademark of Novell, Inc.

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

LANalyzer Agent is a trademark of Novell, Inc.

Link Support Layer and LSL are trademarks of Novell, Inc.

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

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

Media Support Module and MSM are trademarks of Novell, Inc.

Mirrored Server Link and MSL are trademarks of Novell, Inc.

Mobile IPX is a trademark of Novell, Inc.

Multiple Link Interface and MLI are trademarks of Novell, Inc.

Multiple Link Interface Driver and MLID are trademarks of Novell, Inc.

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

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

Natural Language Interface for Help is a trademark of Novell, Inc.

NDS is a trademark of Novell, Inc.
NDS Manager is a trademark of Novell, Inc.
NE/2 is a trademark of Novell, Inc.
NE/2-32 is a trademark of Novell, Inc.
NE/2T is a trademark of Novell, Inc.
NE1000 is a trademark of Novell, Inc.
NE1500T is a trademark of Novell, Inc.
NE2000 is a trademark of Novell, Inc.
NE2000T is a trademark of Novell, Inc.
NE2100 is a trademark of Novell, Inc.
NE21500T is a trademark of Novell, Inc.
NE3200 is a trademark of Novell, Inc.
NE32HUB is a trademark of Novell, Inc.
NEST is a trademark of Novell, Inc.
NEST Autoroute is a trademark of Novell, Inc.
NetExplorer is a trademark of Novell, Inc.
NetNotes is a registered trademark of Novell, Inc. in the United States and other countries.
NetSync is a trademark of Novell, Inc.
NetWare is a registered trademark of Novell, Inc. in the United States and other countries.
NetWare 3 is a trademark of Novell, Inc.
NetWare 3270 CUT Workstation is a trademark of Novell, Inc.
NetWare 3270 LAN Workstation is a trademark of Novell, Inc.
NetWare 386 is a trademark of Novell, Inc.
NetWare 4 is a trademark of Novell, Inc.
NetWare 5 is a trademark of Novell, Inc.
NetWare Access Server is a trademark of Novell, Inc.
NetWare Access Services is a trademark of Novell, Inc.
NetWare Application Manager is a trademark of Novell, Inc.
NetWare Application Notes is a trademark of Novell, Inc.
NetWare Asynchronous Communication Services and NACS are trademarks of Novell, Inc.
NetWare Asynchronous Services Interface and NASI are trademarks of Novell, Inc.
NetWare Aware is a trademark of Novell, Inc.
NetWare Basic MHS is a trademark of Novell, Inc.
NetWare BranchLink Router is a trademark of Novell, Inc.
NetWare Care is a trademark of Novell, Inc.
NetWare Communication Services Manager is a trademark of Novell, Inc.
NetWare Connect is a registered trademark of Novell, Inc. in the United States.
NetWare Core Protocol and NCP are trademarks of Novell, Inc.
NetWare Distributed Management Services is a trademark of Novell, Inc.
NetWare Document Management Services is a trademark of Novell, Inc.
NetWare DOS Requester and NDR are trademarks of Novell, Inc.
NetWare Enterprise Router is a trademark of Novell, Inc.
NetWare Express is a registered service mark of Novell, Inc. in the United States and other countries.

NetWare Global Messaging and NGM are trademarks of Novell, Inc.
NetWare Global MHS is a trademark of Novell, Inc.
NetWare HostPrint is a registered trademark of Novell, Inc. in the United States.
NetWare IPX Router is a trademark of Novell, Inc.
NetWare LANalyzer Agent is a trademark of Novell, Inc.
NetWare Link Services Protocol and NLSP are trademarks of Novell, Inc.
NetWare Link/ATM is a trademark of Novell, Inc.
NetWare Link/Frame Relay is a trademark of Novell, Inc.
NetWare Link/PPP is a trademark of Novell, Inc.
NetWare Link/X.25 is a trademark of Novell, Inc.
NetWare Loadable Module and NLM are trademarks of Novell, Inc.
NetWare LU6.2 is trademark of Novell, Inc.
NetWare Management Agent is a trademark of Novell, Inc.
NetWare Management System and NMS are trademarks of Novell, Inc.
NetWare Message Handling Service and NetWare MHS are trademarks of Novell, Inc.
NetWare MHS Mailslots is a registered trademark of Novell, Inc. in the United States and other countries.
NetWare Mirrored Server Link and NMSL are trademarks of Novell, Inc.
NetWare Mobile is a trademark of Novell, Inc.
NetWare Mobile IPX is a trademark of Novell, Inc.
NetWare MultiProtocol Router and NetWare MPR are trademarks of Novell, Inc.
NetWare MultiProtocol Router Plus is a trademark of Novell, Inc.
NetWare Name Service is a registered trademark of Novell, Inc. in the United States and other countries.
NetWare Navigator is a trademark of Novell, Inc.
NetWare Peripheral Architecture is a trademark of Novell, Inc.
NetWare Print Server is a trademark of Novell, Inc.
NetWare Ready is a trademark of Novell, Inc.
NetWare Requester is a trademark of Novell, Inc.
NetWare Runtime is a trademark of Novell, Inc.
NetWare RX-Net is a trademark of Novell, Inc.
NetWare SFT is a trademark of Novell, Inc.
NetWare SFT III is a trademark of Novell, Inc.
NetWare SNA Gateway is a trademark of Novell, Inc.
NetWare SNA Links is a trademark of Novell, Inc.
NetWare SQL is a trademark of Novell, Inc.
NetWare Storage Management Services and NetWare SMS are trademarks of Novell, Inc.
NetWare Telephony Services is a trademark of Novell, Inc.
NetWare Tools is a trademark of Novell, Inc.
NetWare UAM is a trademark of Novell, Inc.
NetWare WAN Links is a trademark of Novell, Inc.
NetWare/IP is a trademark of Novell, Inc.
NetWare is a registered service mark of Novell, Inc. in the United States and other countries.

Network Navigator is a registered trademark of Novell, Inc. in the United States.

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

Network Navigator - Dispatcher is a registered trademark of Novell, Inc. in the United States.

Network Support Encyclopedia and NSE are trademarks of Novell, Inc. Network Support Encyclopedia Professional Volume and NSEPro are trademarks of Novell, Inc.

NetWorld is a registered service mark of Novell, Inc. in the United States and other countries.

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

Novell Academic Education Partner and NAEP are service marks of Novell, Inc.

Novell Alliance Partners Program is a collective mark of Novell, Inc.

Novell Application Launcher is a trademark of Novell, Inc.

Novell Application Notes is a trademark of Novell, Inc.

Novell Authorized CNE is a trademark and service mark of Novell, Inc.

Novell Authorized Education Center and NAEC are service marks of Novell, Inc.

Novell Authorized Partner is a service mark of Novell, Inc.

Novell Authorized Reseller is a service mark of Novell, Inc.

Novell Authorized Service Center and NASC are service marks of Novell, Inc.

Novell BorderManager is a trademark of Novell, Inc.

Novell BorderManager FastCache is a trademark of Novell, Inc.

Novell Client is a trademark of Novell, Inc.

Novell Corporate Symbol is a trademark of Novell, Inc.

Novell Customer Connections is a registered trademark of Novell, Inc. in the United States.

Novell Directory Services and NDS are trademarks of Novell, Inc.

Novell Distributed Print Services and NDPS are trademarks of Novell, Inc.

Novell ElectroText is a trademark of Novell, Inc.

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

NEST is a trademark of Novell, Inc.

Novell Gold Authorized Reseller is a service mark of Novell, Inc.

Novell Gold Partner is a service mark of Novell, Inc.

Novell Labs is a trademark of Novell, Inc.

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

Novell NE/2 is a trademark of Novell, Inc.

Novell NE/2-32 is a trademark of Novell, Inc.

Novell NE3200 is a trademark of Novell, Inc.

Novell Network Registry is a service mark of Novell, Inc.

Novell Platinum Partner is a service mark of Novell, Inc.

Novell Press is a trademark of Novell, Inc.

Novell Press Logo (teeth logo) is a registered trademark of Novell, Inc. in the United States and other countries.

Novell Replication Services is a trademark of Novell, Inc.

Novell Research Reports is a trademark of Novell, Inc.

Novell RX-Net/2 is a trademark of Novell, Inc.

Novell Service Partner is a trademark of Novell, Inc.

Novell Storage Services is a trademark of Novell, Inc.

Novell Support Connection is a trademark of Novell, Inc.

Novell Technical Services and NTS are service marks of Novell, Inc.

Novell Technology Institute and NTI are registered service marks of Novell, Inc. in the United States and other countries.

Novell Virtual Terminal and NVT are trademarks of Novell, Inc.

Novell Web Server is a trademark of Novell, Inc.

Novell World Wide is a trademark of Novell, Inc.

NSE Online is a service mark of Novell, Inc.

NTR2000 is a trademark of Novell, Inc.

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

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

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

Open Data-Link Interface and ODI are trademarks of Novell, Inc.

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

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

Open Socket is a registered trademark of Novell, Inc. in the United States.

Packet Burst is a trademark of Novell, Inc.

PartnerNet is a trademark and service mark of Novell, Inc.

PC Navigator is a trademark of Novell, Inc.

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

Perform3 is a trademark of Novell, Inc.

Personal NetWare is a trademark of Novell, Inc.

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

Portable NetWare is a trademark of Novell, Inc.

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

Print Managing Agent is a trademark of Novell, Inc.

Printer Agent is a trademark of Novell, Inc.

QuickFinder is a trademark of Novell, Inc.

Red Box is a trademark of Novell, Inc.

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

Remote Console is a trademark of Novell, Inc.

Remote MHS is a trademark of Novell, Inc.

RX-Net is a trademark of Novell, Inc.
RX-Net/2 is a trademark of Novell, Inc.
ScanXpress is a registered trademark of Novell, Inc. in the United States and other countries.
Script Director is a registered trademark of Novell, Inc. in the United States and other countries.
Sequenced Packet Exchange and SPX are trademarks of Novell, Inc.
Service Response System is a trademark of Novell, Inc.
Serving FTP is a trademark of Novell, Inc.
SFT is a trademark of Novell, Inc.
SFT III is a trademark of Novell, Inc.
SoftSolutions is a registered trademark of SoftSolutions Technology Corporation, a wholly owned subsidiary of Novell, Inc.
Software Transformation, Inc. is a registered trademark of Software Transformation, Inc., a wholly owned subsidiary of Novell, Inc.
SPX/IPX is a trademark of Novell, Inc.
StarLink is a registered trademark of Novell, Inc. in the United States and other countries.
Storage Management Services and SMS are trademarks of Novell, Inc.
Technical Support Alliance and TSA are collective marks of Novell, Inc.
The Fastest Way to Find the Right Word is a registered trademark of Novell, Inc. in the United States and other countries.
The Novell Network Symbol is a trademark of Novell, Inc.
Topology Specific Module and TSM are trademarks of Novell, Inc.
Transaction Tracking System and TTS are trademarks of Novell, Inc.
Universal Component System is a registered trademark of Novell, Inc. in the United States and other countries.
Virtual Loadable Module and VLM are trademarks of Novell, Inc.
Writer's Workbench is a registered trademark of Novell, Inc. in the United States and other countries.
Yes, It Runs with NetWare (logo) is a trademark of Novell, Inc.
Yes, NetWare Tested and Approved (logo) is a trademark of Novell, Inc.
Yes, Tested and Approved is a trademark of Novell, Inc.
Z.E.N.works is a trademark of Novell, Inc.

Third-Party Trademarks

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

