# Solstice Backup™ 5.5 Administration Guide

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303
U.S.A. 650-960-1300

Part No. 806-1135-10
September 1999, Revision A

# Contents

# Figures

# Tables

# Preface

The *Solstice Backup 5.5 Administration Guide*, contains information about how to configure and manage the Sun Backup for Solaris™ software.

This guide also provides information about how to use and administer the Backup software when it is installed in a typical networked environment. For more detailed technical information about Backup commands, refer to the online Backup manual (man) pages after you install the software.

## Audience

The information in this guide is intended for system administrators who are responsible for installing software and maintaining the servers and clients on a network. Operators who monitor the daily backups may also find this manual useful.

## About This Guide

You *must* install the Backup software on your server and clients to use the information presented in this guide. If you have not yet installed the software, refer to the *Solstice Backup 5.5 Installation Guide and Release Notes* version for your operating system for installation instructions.

The *Administrator's Guide* contains detailed information about how to set up, configure, and use the Backup software, backup devices, and backup media. This guide assumes you have a basic knowledge of networks, backup devices, and other technical subjects related to computer hardware, software, and networks necessary for the installation, configuration, and use of the Backup product.

A portable document format (PDF) version of *Solstice Backup 5.5 Administration Guide* is included with the Backup software. Refer to *Solstice Backup 5.5 Installation Guide and Release Notes* for information and instructions about how to access the PDF files and install the Adobe® Acrobat Reader on your machine.

# How This Book Is Organized

This book is organized as follows:

**Chapter 1, "Introduction,"** contains information about the features provided by your Solstice Backup for UNIX software and a suggested roadmap to follow to configure and learn how to use the Backup software.

**Chapter 2, "Getting Started,"** explains in more detail the configuration tasks and Quick Test described in the *Solstice Backup 5.5 Installation Guide and Release Notes*.

**Chapter 3, "Server and Storage Node Operations,"** describes operations that you manage through the Backup server.

**Chapter 4, "Device and Media Management,"** describes device and media operations you can perform through the Backup server.

**Chapter 5, "Client Operations,"** describes how to configure and use Backup clients, and gives suggestions on how to best customize your client configurations to suit the needs of your environment.

**Chapter 6, "Archive,"** explains how to install and use the optional Backup archive application.

**Chapter 7, "Autochangers,"** provides information about how to install, configure, and operate Backup autochanger support.

**Chapter 8, "Hierarchical Storage Management,"** includes instructions to configure the Solstice Backup HSM or Solstice Backup HSM XDSM software on clients of the Backup server.

**Chapter 9, "SmartMedia,"** provides information about how to install, configure, and operate Backup autochanger support.

**Chapter 10, “SNMP Module,”** provides instructions to configure and use the Simple Network Management Protocol (SNMP) Module from Solstice SunNet Manager™ or HP OpenView Network Node Management window.

**Chapter 11, “Silos,”** describes the Silo Support Module that you can use with Backup NetWork Edition or Backup Power Edition.

**Appendix A, “Backup Functionality,”** provides a brief, simplified overview of how Backup performs a backup and recovery.

**Appendix B, “Command Line Reference Utilities,”** provides an abbreviated reference for some of the options available through the command line interface.

**Appendix C, “Troubleshooting,”** gives information that helps diagnose problems.

# Other Backup Documentation

The *Solstice Backup 5.5 Administration Guide* is part of a complete documentation suite, consisting of electronic versions of the documentation as well as online program help and online man pages. A PDF version of each of these documents is included with the Backup software.

## Installation Guide

The *Solstice Backup 5.5 Installation Guide and Release Notes* is your single source for installation information, such as:

- How to install the software components on your server, client, and designated storage node machines
- How to configure your backup devices
- How to update or upgrade your current Backup software
- How to install the Acrobat Reader and access the PDF versions of the printed manuals
- How to remove the software components

Read the *Solstice Backup 5.5 Installation Guide and Release Notes* thoroughly before you install the Backup software. After the installation is completed, keep a printed copy in a safe place so it is available if you need to reinstall or remove the software later.

The *Solstice Backup 5.5 Installation Guide and Release Notes* for the Solstice Backup ClientPak for NetWare, UNIX, and Windows Clients software provides instructions for installing the client software on other operating systems that are not supported as Backup servers but *are* supported as Backup clients.

## Disaster Recovery Guide

The *Solstice Backup 5.5 Disaster Recovery Guide* is a multiplatform guide that provides information about how to use Backup products to recover data from your servers in case of a disaster. This guide also contains tips and information about preventive measures to safeguard data and prepare your backup policies to guard against a disaster before one strikes.

Review this guide when you first install and configure Backup to be prepared with strategies to protect your data and operations in case of a disaster. Keep a printed copy of the guide with your other documents regarding disaster recovery so that it is available in case of a disaster.

A PDF version is included with the Backup software distribution. Refer to the *Solstice Backup 5.5 Installation Guide and Release Notes* for information and instructions on how to access the PDF files and install the Acrobat Reader.

## Online Help

The Backup graphical user interface (GUI) includes an online help component. The online help contains specific information about the GUI, including context-sensitive help that describes the purpose of each window and what information to enter in each field. There are also help topics that describe Backup features and topics that provide step-by-step procedures for doing Backup tasks with the GUI.

To access the online help, select the Help menu displayed in the GUI. You have four choices:

- On Window provides help on the current window.
- On Topic provides a list of help topics to choose from.
- On Help provides information on how to use the online help.
- On Version provides information on the version of Backup software that is installed on your system.

Backup also provides Help buttons in most of the dialog boxes. Clicking the Help button in a Backup dialog box displays the help topic associated with that Backup feature.

# Online Manual Pages

The manual (man) pages are a reference for the syntax and function of the Backup commands you issue from the command line. To view a man page, make sure that the MANPATH environment variable includes the path where you installed the Backup man pages, then enter the `man` command plus the name of the command at the shell prompt. For example, to view the man page for the `nsrjb` command, type:

```
% man nsrjb
```

To print a copy of the entire collection of Backup man pages, enter the `troff` command at the shell prompt with the options shown in this example:

```
% troff -t -man ‘nsr_man -l’| lpr -t -P printer-name
```

The command for your machine may vary (for example, your print command may be `lp` instead of `lpr`), depending on the operating system and the version of PostScript or troff software you have installed.

# What Typographic Changes Mean

The following table describes the typographic changes used in this book.

**TABLE P-1** Typographic Conventions

| Typeface or Symbol | Meaning | Example |
|---|---|---|
| `AaBbCc123` | The names of commands, files, and directories; on-screen computer output | Edit your `.login` file.<br>Use `ls -a` to list all files.<br>`machine_name% You have mail.` |
| **`AaBbCc123`** | What you type, contrasted with on-screen computer output | `machine_name%` **`su`**<br>`Password:` |
| *AaBbCc123* | Command-line placeholder: replace with a real name or value | To delete a file, type `rm` *filename*. |
| *AaBbCc123* | Book titles, new words or terms, or words to be emphasized | Read Chapter 6 in *User's Guide*. These are called *class* options.<br>You *must* be root to do this. |

# Shell Prompts in Command Examples

The following table shows the default system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

**TABLE P-2** Shell Prompts

| Shell | Prompt |
|---|---|
| C shell prompt | `machine_name%` |
| C shell superuser prompt | `machine_name#` |
| Bourne shell and Korn shell prompt | `$` |
| Bourne shell and Korn shell superuser prompt | `#` |

CHAPTER 1

# Introduction

This chapter contains information about the features provided by your Solstice Backup software and a suggested roadmap to follow to configure and learn how to use the Backup software.

This chapter consists of the following sections:

- What is Backup?
- How Backup Works
- Software License Modes
- Roadmap of Backup Configuration Tasks

## What Is Backup?

It is important to back up the data on computer systems to prevent the loss of valuable data. In a networked environment, where users depend on shared data and the amount of data grows rapidly as systems are added to the network, the need to manage data becomes crucial.

Solstice Backup is a solution for network storage management and backup; it protects and helps manage data across an entire network of computers. Backup saves valuable administrator time by speeding up and simplifying daily operations of storage management. A graphical user interface (GUI) guides you through administering and configuring your network for storage management. As the Backup software manages your data, it creates a database of saved data, making it easy to locate data for recovery. Furthermore, as the network and amount of data grow, Backup provides the capacity and performance to handle the load.

Backup software features include a storage management application that directs high performance to a wide range of storage devices. The Backup software product is made up of the following components:

- Backup client software, which is a collection of processes and programs installed on the computers that contain data to be managed
- Backup server software, which is a collection of processes and programs installed on the computer that performs the Backup services, including data backup, recovery, archive, and Backup HSM (Hierarchical Storage Management)
- Backup storage node software, which is a collection of processes and programs installed on computers that control their attached storage devices during Backup operations, including backup, recovery, archive, and HSM

You can configure your Backup software to back up data to storage devices that are directly connected to the Backup server, or you can create a Backup *storage node,* which is a separate computer with one or more storage devices connected. The Backup server maintains the resources and files that keep track of Backup data. The storage node computer controls the storage devices and media.

You can direct backup data, archive data, and migrate data to specific collections of volumes, according to particular data characteristics. As data is written to storage media, the Backup server creates a database to track the location of all managed data. The Backup server monitors data status and automatically rotates data through the tracking system as it reaches different levels of obsolescence, as defined by you.

You and your users can browse an index of recoverable files, create reports that describe the status of data or media, and recover data to a point in time specified by the user. In response to a data recovery request, the Backup storage management system locates the volume that contains the requested data and either directs a device to mount the appropriate volume for retrieval, or sends a message to the operator to request the volume by name. In this way, Backup manages all storage volume operations.

## Cross-Platform Management

Backup is a cross-platform product for storage management. You can administer a Backup server from a workstation running UNIX, Windows NT, or Novell NetWare, if the workstation has the Backup server software installed and is connected by a network. Additionally, a Backup client on one platform can back up its data to a Backup server of a different platform.

The Backup server can direct and conduct administration services for any client or server on the network that has the Backup software installed and is recognized by the Backup server. The Backup administration program interface offers you only the options that are relevant to a particular Backup client, so you do not need to know the configuration of individual computers before you start a monitoring session.

# Performance

Backup has many standard and optional performance features:

- Parallelism, which allows several *savestreams* to flow to the server or storage node at the same time
- Multiplexing, which lets more than one savestream write to the same device at the same time
- Client parallelism, which lets the client send more than one savestream at a time
- Session management, which enables you to control the number of savestreams per device to maximize the performance of each device
- Backup to file devices and optional subsequent staging to nearline or offline volumes
- Optional Backup software additions, such as:
  - Backup Autochanger Module
  - Backup Silo Software Module
  - Backup Archive
  - Backup BusinesSuite Module
  - Backup SNMP (Simple Network Management Protocol)
  - Backup SmartMedia
  - Remote Library Managers (RLM)
  - Backup HSM
  - Cluster server and client support (only available with Backup PowerEdition™)

# Ease of Use

Backup provides tools to make protection of critical data easy to manage. With Backup, you can:

- Use either the Backup GUI or the command line to manage Backup
- Administer and configure Backup functions from any computer on the network
- Grant permission to provide the capability for recovery of one client’s data to another client computer of the same operating system
- Obtain immediate answers to questions by accessing online help and man pages
- Take advantage of the automatic media management feature to allow the Backup server or storage node to label and mount volumes as needed for backups
- Use the Tech Dialog and technical bulletins on the Sun web site to find answers to common questions

## Scalability

You can add to your current Backup software purchase as your storage management needs grow. For example, you can:

- Upgrade the basic level of server functionality, add support for additional (or larger) autochangers, add support for more clients, or add optional software modules without the need to reinstall the server software.
- Add special BusinesSuite Module client software to back up databases and other nonfilesystem data.
- Add support for remote storage nodes to control backup devices, while the data management tasks remain centralized on a controlling Backup server.
- Introduce Backup GEMS into your enterprise environment to streamline storage management and provide comprehensive administrative policies that control media, devices, software, licensing, and Backup servers and clients. The GEMS Java-based interface allows you to administer, configure, and monitor your storage management applications from one location.

# Backup Product Family

Sun has a large product family of server, client, database, and related backup-and-recover software.

## Server Software Products

All Backup clients (and servers, which can function as their own clients and back up their own data) can be backed up by Backup server software, which runs on the following platforms:

- UNIX (AIX, HP-UX, IRIX, Dynix/PTX, and Solaris)
- Windows NT
- Novell NetWare/IntranetWare

Backup server products are available in three distinct versions:

- Backup WorkGroup Edition, which enables you to back up the server and up to three clients. Backup WorkGroup Edition does not include support for additional clients or optional software features such as Backup Archive. If you decide later to purchase a higher level of Backup software, all the data backed up by Backup WorkGroup Edition is recoverable by any level of Backup software you purchase.

- Backup Network Edition, which enables you to back up the server and as many client workstations as you purchased client connections for. You can upgrade Backup Network Edition to include support for additional clients, as well as optional software features.
- Backup Power Edition, a server software product that has been specially tuned to protect and provide backup for environments that support VLDB (very large database) or large filesystem applications (in the terabyte range). You can upgrade Backup Power Edition to include support for high-speed devices as well as cluster servers and clients.

## Client Software Products

Some Backup client software products are included with the server products. Refer to the *Solstice Backup 5.5 Installation Guide and Release Notes* for the client support packages available, and for instructions explaining how to install the software. Support for platform-specific clients is also sold separately as Backup ClientPak™ software. Presently, Sun produces client software support for the following platforms and operating systems:

- UNIX: Solaris, SunOS, DIGITAL UNIX, Dynix/PTX, HP-UX, AIX, IRIX, and UnixWare
- PC Desktops: Windows NT Workstation, Windows 98, and Windows 95
- Windows NT Server
- Novell NetWare/IntranetWare
- Macintosh

## Other Products

Sun offers other products to enhance your storage management needs:

- The Backup Autochanger module, which enables you to fully use the automatic backup capabilities of Backup when connected to one or more autochangers
- The Backup Archive application, which provides you with a sophisticated method to archive valuable data for long-term storage
- The Backup HSM and XDSM HSM applications, which provide a method to automatically move data between different media types and locations, thereby conserving network management resources
- The Backup SNMP (simple network management protocol) module, which allows communication of Backup event notifications to SNMP-compliant network management stations

- The Backup BusinesSuite modules, which reliably back up the most widely used databases, including Microsoft Exchange Server and SQL Server™, Oracle, INFORMIX, Lotus Notes, and Sybase

We continue to develop enhancements to the Backup product line, including additional server, client, database, and add-on products. Access the Sun web site (`http://www.sun.com`) for the latest product information.

# How Backup Works

The *Backup server* is the computer on your network that runs the Backup server software, stores the client indexes and media database, and provides backup and recovery services to the clients on the network. You can connect storage devices to the Backup server or attach them to a *storage node*. The *Backup client* must have the client software installed and must be included in the server's list of clients. All server and storage node computers that you want a client to back up to must be listed in the `nsrhost` file, which is created when you install the client software.

The Backup server backs up client data in increments called *save sets*. A save set typically comprises all the backed-up data from a single filesystem or logical disk. Save sets are saved to a *volume* (for example, tapes or optical disk) mounted in a backup device attached to the server. The server uses a *pool* to sort specific data to preselected volumes to help manage your data and collection of volumes.

The Backup server maintains records of the client save sets and the volumes in a *client file index* for each client and a *media database* on the server. The Backup software uses these records to track the saved files and the volumes on which the files are stored. The client file index and media database contain the tracking information for all data controlled by the Backup server, whether the data is written to a device attached to the server or a device attached to a storage node.

When the Backup server backs up files, you might receive a request to mount a writable volume in the server's backup device. When a Backup user recovers files, you receive a request to mount a specific volume by its name. To fulfill either request, you only need to mount the requested volume or volumes in the device attached to the server. If you back up to an autochanger, the requested volume is automatically loaded if it is stored in the autochanger.

This guide uses the term *autochanger* to refer to a variety of robotic libraries, including jukebox, autoloader, carousel, library, nearline storage, and datawheel devices.

The Backup software supports many of the latest backup devices that provide the highest performance and most reliable solutions for your network backup and recovery requirements.

# Software License Modes

The Backup software is distributed on CD-ROM. The CD-ROM contains electronic versions of the Backup documentation.

You can use the software products in one of three software license modes:

- Evaluation mode
- Enabled mode
- Authorized mode

*Evaluation mode* software is distributed on a free, trial basis for 30 days. You can use and evaluate the software without entering an enabler code for 30 days after you install the software on your computer. If you decide that you want to continue to use Backup, you must purchase the appropriate enabler code for all the features you want to keep. If you do not enter the proper enabler codes before the evaluation period ends, the Backup software does not allow further backups or configuration to take place. You can still, however, recover any data that was written to a Backup device during the 30-day evaluation.

If you already have a Backup product installed and enabled and you want to evaluate additional product features, you must enter a special evaluation enabler for each feature. You do not need to enter the special evaluation enabler codes if you are also evaluating the Backup software. Refer to the *Solstice Backup 5.5 Installation Guide and Release Notes* for the evaluation enabler codes for this release of the software. When you enter the special evaluation enabler codes, you can evaluate and use the software for 45 days.

*Enabled mode* requires a code that is included with the purchased product on an *Enabler Certificate*. The code that enables the Backup server is referred to as a *base enabler.* Enabler codes are generic to the product release and are entered on the Backup server. Enabled mode enables you to use the Backup software for 45 days beginning the day you enter the enabler code. After you enter the enabler code for the product, you must register the product to continue to use the software after the 45 days expire. Refer to the *Installation Guide and Release Notes* for instructions about how to print the required product registration document. Your product registration information contains the host ID of the Backup server computer, which Sun incorporates into a unique authorization code to bind the license agreement to a specific computer. If you do not register the product and then obtain and enter the authorization code during the 45-day enabled period, the Backup software does not allow further backups or configuration to take place. You can still recover previously backed-up data after the software expires.

*Authorized mode* begins when you enter the authorization code provided by Sun. After you enter the authorization code for your Sun products, the software is available for permanent use for the given computer. If you need to transfer the software to another computer (or transfer the computer to another IP address), you must get a *Host Transfer Affidavit* from Sun Customer Service and submit the form to Sun. If you do not receive new authorization codes after you move the Backup software, the software disables itself in 15 days.

---

**Caution –** You must remove the server software from the computer you transferred the software from or shut down the `nsrd` daemon on that computer before you start the daemon on the other computer. If you start the `nsrd` daemon on another computer with the same base enabler, you receive a copy violation error message and the software is disabled.

---

# Roadmap of Backup Configuration Tasks

This section provides a suggested roadmap for you to follow to set up and configure Backup for your environment. Cross-references to instructions found elsewhere in this *Administrator's Guide* are included. If you view the PDF version of this guide, click on the cross-reference text to go to the information.

1. **Install and enable the Backup software according to the instructions in the** *Installation Guide and Release Notes***.**

2. **Configure the Backup server. See "Basic Configuration for the Backup Server" on page 11 for more information.**

3. **Configure one or more devices:**
   - For stand-alone devices, see "Device Configuration" on page 71.
   - For devices in an autochanger, see "Autochanger Device Configuration" on page 73. See "Autochanger Configuration and Management" on page 163 for a complete discussion about how to manage an autochanger for use with the Backup software.
   - For devices in a silo, see "Silo Installation and Configuration" on page 238.
   - For file devices, see "Device Configuration" on page 71 and "Save Set Staging" on page 93.

4. **Configure the client resource for the server and any additional clients that connect to the server. See "Clients Resource" on page 16.**

5. **Register your Backup software and enter the authorization code returned to you within 45 days of enabling it. See "How to Register and Authorize Your Software" on page 13 for more information.**

The Backup software includes default configurations that allow you to back up data without further configuration. The default configuration values are described in “Getting Started” on page 11. Chapter 2 also provides the basic information you need to configure and use your Backup software.

After you become familiar with Backup and your storage management needs expand beyond the coverage of the default configurations, you can customize the following components:

- Backup groups:

  You can create a scheduled backup group, to spread the backup task load on your server, and assign the appropriate clients to the group. See “Backup Group Configuration” on page 34 for instructions about configuring a group and “How to Create a New Client” on page 102 for client configuration.

- Pools and label templates:

  You can configure a Backup pool for a backup group to associate with, to segregate backed-up data according to its characteristics, such as which client it comes from or what level of backup it is from. See “Pools” on page 76 for information about how to create custom pools to hold the segregated data and “Customizing Label Templates” on page 83 to configure the internal labels that are placed on the volumes.

- Backup schedules and policies:

  You can customize backup levels to help manage the number of volumes required for backups and recoveries. To customize a backup cycle with defined backup levels, see “How Backup Levels Work” on page 47. See “Customizing Policies” on page 120 for instructions on how to define customized policies for save set browsing and retention to help manage the data life cycle.

- Customized directives:

  You can define a customized directive that applies a specific set of instructions to a client’s save sets. See “How to Create Customized Directives” on page 123 for information about how to define a customized directive to apply to the appropriate client resource.

- Notifications:

  You can receive information about the Backup server’s activities, such as software registration reminders, index size alerts, tape mount requests, and completion of scheduled backups. For further information, see “Event Notification” on page 64.

CHAPTER 2

# Getting Started

This chapter explains in more detail the configuration tasks and Quick Test described in the *Solstice Backup 5.5 Installation Guide and Release Notes*. It describes the initial administration tasks required to get the Backup software running with a default configuration and lists the default configurations for major Backup components. This chapter consists of the following sections:

- Initial Administration Tasks
- Preconfigured Resources

## Initial Administration Tasks

After you install the software according to the instructions in the *Solstice Backup 5.5 Installation Guide and Release Notes*, you must do several tasks before you can back up data.

### Basic Configuration for the Backup Server

To configure your Backup server, you can either use the graphical version (`nwadmin`) or character-based interface (`nsradmin`) of the Backup administration program, or enter commands at the command line. Refer to the online help or Backup man pages for instructions explaining how to use each interface. Before you can run the Backup software, some configuration is required that is strictly unique to your environment:

- Enter an enabler code that unlocks the features of the software that you purchased to use for 45 days. (Evaluation enabler codes unlock the features of the software for you to use for 45 days. Before the 45 days expire, you must purchase and enter a permanent enabler code to continue to use each evaluated feature).

- Register the software. Enter the authorization code that Sun returns to permanently enable your products.

- Configure your devices and label the backup media.
- Set up the client portion of the Backup server and other Backup clients.

Only users who have administrative privileges can create or apply configurations. Because only `root@server-`*name* has Backup administrative privileges by default, you must become root on the Backup server before you start any Backup administration programs or tasks. You can add other users to the list of authorized Backup administrators later.

The Backup server manages the clients on a network through a resource allocation protocol (RAP). The server maintains the configurations that you enter as *resources*. Each resource contains a set of *attributes* to which values are assigned (for example, the Name attribute found in several Backup resources contains a value that defines the name of the resource).

The GUI analogy for a Backup resource is a window; the analogy for a Backup attribute is a field, button, or check box in the window.

You can configure and manage your Backup software in the following ways:

- Backup administration program (`nwadmin`)

You can register, configure, and monitor the Backup servers, clients, storage nodes, and devices in your network using the Backup administration program. The Backup administration program is a GUI for X Windows environments. To start the Backup administration program, type `nwadmin` at the shell prompt.

- Backup character-based interface (`nsradmin`)

You can use the Backup character-based interface on any display, including those that do not support graphics. You can perform many of the same configuration and management tasks as in the GUI. To start the character-based version of the Backup administration program, type `nsradmin -c` at the shell prompt. For more information about using the character-based interface, refer to the `nsradmin` man page.

- Backup command line interface

You can perform any of the configuration and administration tasks available through the Backup administration program, as well as the tasks available in the client-side backup, recover, archive, and retrieve programs, through Backup commands that you enter at the shell prompt. See “Command Line Reference Utilities” on page 261 and the referenced online man pages for instructions on how to use the Backup command line interface.

## Enabler Code Entry

After you install the Backup server software, you can use either the Backup administration program (`nwadmin`) or the `nsrcap` command to enter the *enabler codes* you purchased for your Backup software products. The enabler code is printed on the *Enabler Certificate* you received. Enter the base enabler code for the Backup server before you enter the enabler codes for additional features that you purchased, such as the Autochanger Software Module.

To enter your enabler code, become root on the Backup server and follow these instructions:

1. **Issue the following command to start the GUI version of the Backup administration program:**

```
# nwadmin -s server-name &
```

2. **Open the Registration window.**
3. **Click Create.**
4. **Enter the enabler code.**
5. **Click Apply.**

After you enter the enabler code for the software, you can configure and perform Backup backups for up to 45 days before you must enter an *authorization code* to continue to use the Backup software on a permanent basis.

You can purchase additional enablers for added features, such as Backup Archive and Backup HSM as well as support for additional clients or autochangers that you add to your network after you install the Backup software.

## ▼ How to Register and Authorize Your Software

After you enable your Backup products, you must register and authorize the products within 45 days to continue to perform backups. To register your software, follow these steps:

1. **Become root on the system where you installed the Backup server software.**
2. **Enter the** `nwadmin` **command or the** `nsradmin -c` **command at the shell prompt to start the Backup administration program.**
3. **Enter your company and product information in the Server window (**`nwadmin`**) or select and edit the Backup Save and Recover (NSR) License resource (**`nsradmin`**). You must enter all required company and product information in the Server resource, or you cannot register your products properly.**
4. **Change the Server window to a tabular view and print the contents of the Server resource.**

5. **Mail or fax the printout to Sun Customer Service.**

6. **When you receive the authorization code from Sun, become root on the Backup server and start the administration program.**

7. **Open the Registration window, enter the authorization code in the Auth Code field, and apply the changes. Repeat the entry process for each Backup product you purchased.**

   After you enter the authorization code, your Backup software is authorized for permanent use.

## ▼ How to Remove an Enabler Code

If you need to remove an enabler code later, whether or not it is permanently authorized, follow these steps:

1. **Become root on the system where you installed the Backup server software.**

2. **Enter the `nwadmin` command at the shell prompt to start the Backup administration program, or enter the `nsradmin -c` command to start the character-based interface in visual mode.**

3. **Open the Registration window (`nwadmin`) or edit the NSR_license (`nsradmin`) resource.**

4. **Highlight the enabler code you want to remove and select the Delete option.**

---

**Caution –** You *cannot* remove a *base enabler code*. You *can* update a base enabler, using the `nsrcap -v -u` command.

---

## Device Configuration

Before the Backup server can use your storage devices, you must first configure each storage device separately. A device is either stand alone, a file, or located in an autochanger or silo tape library (STL). The type of device (stand alone, file, autochanger, or silo) determines how you configure the device.

When you install the Backup software, Backup configures the SCSI devices for the device choices you make. If you decide you need to add, delete, or modify your Backup storage devices after you install Backup, edit the Devices or Jukeboxes resources. You can use either the `nwadmin` (GUI) or the `nsradmin` (character-based) version of the Backup administration program.

**Caution –** For devices in STLs, you cannot use nwadmin or nsradmin to add, delete, or modify the configuration. See Chapter 11, "Silos," on page 237 for more information about configuring devices in a silo.

For instructions on how to use the nwadmin program or the nsradmin program to configure your backup devices, refer to the online help provided in each program.

### Devices Resource

Use the Devices resource to configure stand alone devices. Enter the name of the device and the media type. Any devices you add in the Devices resource are displayed in the Backup administrator program in the Devices display. For details on device configuration, see "Device Configuration" on page 71.

### Jukeboxes Resource

You perform most of the necessary autochanger configuration with the jb_config command described in the *Solstice Backup 5.5 Installation Guide and Release Notes*. After the jb_config command is completed and exits successfully, the configured options are displayed in the Jukeboxes resource of the administration program. The Jukeboxes resource displays the name of the autochanger, the model, control port pathnames, device pathnames, barcode label choices, and available slots that you originally created when you installed Backup.

The only changes you can make to your autochanger in the Jukeboxes resource are to the available slots, devices, and barcode label attributes. If you need to make additional changes, you must first delete the autochanger from the list, and then run the jb_config command again. For further details on these attributes, see "Autochanger Device Configuration" on page 73.

If your storage devices reside in a silo, see Chapter 11, "Silos," on page 237 for information on the use of a silo with Backup.

## Volume Labels

Before the Backup software can use a volume, you must first label it with a valid Backup label. Backup uses an internal volume label to identify the media it uses for backups, archives, and migration. Backup provides several preconfigured label templates for you to use to label your volumes. See "Label Template Resource" on page 25 for the preconfigured label template settings.

To label a volume, you must first mount it on a device. You can select the Mount speedbar button displayed in the main window of the `nwadmin` program GUI, or you can issue the `nsrmm -m` command at the shell prompt.

# Preconfigured Resources

The software ships with several resources already configured so you can use Backup right away, without need for further configuration. As you become familiar with the software, you can customize the Backup resources to suit your storage management needs.

## Clients Resource

The first client of the Backup server is the server itself. The Client resource for the server is automatically created for you when you install the software. The configurations that ship with Backup are already in place for the client setup for the server. The following table lists the preconfigured settings for the Client resource.

**TABLE 2-1** Client Resource Preconfigured Attribute Settings

| Attribute | Preconfigured Setting |
|---|---|
| Name | Hostname of the Backup server |
| Server | Hostname of the Backup server |
| Archive Services | Disabled (client cannot perform archives) |
| Schedule | Default |
| Browse Policy | Month (client file index entries remain browsable for one month after backup) |
| Retention Policy | Year (retains media database entries for the client’s save sets for one year) |
| Directive | Null (does not use special directives for the client’s backup) |
| Group | Default |
| Save Set | All (backs up all filesystems for the client) |
| Remote Access, Remote User, Password | Null (only users on the client computer can browse the client’s file index and recover data) |

**TABLE 2-1** Client Resource Preconfigured Attribute Settings *(Continued)*

| Attribute | Preconfigured Setting |
|---|---|
| Backup Command | Null (uses the standard client `save` program) |
| Aliases | Null (the client does not have other qualified names used for contact) |
| Archive Users | Null (there are no authorized login IDs required for Archive services) |
| Storage Nodes | `nsrserverhost` (the primary Backup server) |
| Clone Storage Nodes | Null |
| Client OS Type | Varies, depending on the type of operating system (this is a read-only attribute) |
| CPUs | Null (this is a read-only attribute) |
| Backup Version | Null (this is a read-only attribute) |
| Enabler in Use | No (the client is not currently using a license; this is a read-only attribute) |

You can change any of the preconfigured client settings to suit your needs. If you want to apply customized settings for the Group, Schedule, Browse Policy, Retention Policy, or Directive attributes, you must first create customized resources for these attributes before you can apply them to a new or existing Backup client. Because these resources rely on other settings, you must create customized resources in the following order:

- Groups
- Pools
- Schedules
- Policies
- Directives

For information on how to customize resources for Backup clients, see Chapter 5, “Client Operations” on page 101.

After you create the customized resources, all the preconfigured and custom configurations created in the Groups, Schedules, Policies, and Directives resources are displayed as choices in the Clients resource for you to apply to your new or existing clients.

## Groups Resource

In the Groups resource, you determine which computers are backed up together and at what time. Use groups to back up selected computers at different times to control the amount of traffic on your network. You can also use the Groups resource to automatically *clone* backup data.

All Backup clients are initially assigned to the preconfigured Default group. When you first install Backup, the Default group's Autostart attribute is disabled. When you are ready to test your Backup software, you can select Start Now to override the assigned Start Time of 3:33. To begin regularly scheduled backups, select the Enabled choice for Autostart.

The following table lists preconfigured attribute settings for the Groups resource.

**TABLE 2-2** Preconfigured Attribute Settings for the Groups Resource

| Attribute | Preconfigured Setting |
|---|---|
| Name | Default |
| Autostart | Disabled |
| Autorestart | Disabled |
| Stop Now | False |
| Start Time | 3:33 |
| Last Start | Null |
| Interval (how frequently the group runs, using the 24 hour clock) | 24:00 |
| Force Incremental | Yes (if two level Full backups start within 24 hours, force the second one to an incremental level backup) |
| Client Retries | 1 |
| Clones | No |
| Clone Pool | Default Clone |
| Migration Clone Pool | Migration Clone |
| Options | Null |
| Inactivity Timeout (number of minutes of inactivity before concluding the client is hung) | 30 |
| Printer | Assigned default printer for server |

For further details on how to use backup groups, see "Monitoring and Managing Group Backups" on page 38.

## Pools Resource

The Pools resource determines where backup data is directed. Backup ships with several preconfigured pools to choose from. The Default group is preconfigured to be backed up to media labeled for the Default pool. The following table lists preconfigured settings for the pools resource attributes.

**TABLE 2-3** Preconfigured Settings for the Pools Resource Attributes

| Attribute | Preconfigured Setting |
|---|---|
| Name | Default |
| Enabled | Yes |
| Pool Type | Backup |
| Label Template | Default |
| Groups | Default |
| Clients | Null |
| Save Sets | Null |
| Levels | None selected |
| Devices | Device entered during installation |
| Store Index Entries | Yes |
| Auto Media Verify | No |
| Recycle to Other Pools | No |
| Recycle from Other Pools | No |
| Volume Type Preference | Null |

## Schedules Resource

Backup uses the Schedules resource to determine the level of backup for each client on a given calendar day. When you create a new Backup client, the Schedules attribute is automatically assigned the default schedule. You can assign a different preconfigured schedule or customize one of your own.

Backup ships with five preconfigured schedules, described in the followin table. You can use these schedules without further configuration if they fit your backup requirements. Otherwise, you can create new schedules to accommodate your specific needs.

You can modify, but not delete, the preconfigured Default schedule. The attributes for all of the other preconfigured schedules can be deleted or modified, however you cannot modify the names of the preconfigured schedules. The following tablelists preconfigured Backup backup schedules.

**TABLE 2-4** Preconfigured Backup Schedules

| Schedule Name | Backup Backup Operation |
|---|---|
| Default | Completes a full backup every Sunday and incremental backups on all other days. |
| Full on 1st of Month | Completes a full backup on the first calendar day of the month and incremental backups on all other days. |
| Full Every Friday | Completes a full backup every Friday and incremental backups on all other days. |
| Quarterly | Completes a full backup on the first day of each quarter (January, April, July, and October) and performs a level 5 backup on the first day of the other months in the quarter. Every seven days, a level 7 backup occurs; incremental backups occur on all other days.<br>When you customize a quarterly schedule, use the Month period to set the level backups, then use an override to set each quarterly full backup on the calendar. |
| Full on 1st Friday of Month | Completes a full backup on the first Friday of the month and incremental backups on all other days.<br>Backup ships with the *override* options already preset for this schedule. The overrides for this schedule carry over from year to year. |
| Consolidate on 1st of Month | Completes a level 1 backup on the first day of the month and then consolidates that backup with the most recent full backup to produce the consolidated backup. The remaining backups for the month are incrementals. |
| Consolidate 1st Friday of Month | Completes a level 1 backup on the first Friday of the month and then consolidates that backup with the most recent full backup to produce the consolidated backup. The remaining backups for the month are incrementals. |
| Consolidate Every Friday | Completes a level 1 backup every Friday and then consolidates that backup with the most recent full backup to produce the consolidated backup. The remaining backups for the other days of the week are incrementals. |

**TABLE 2-4** Preconfigured Backup Schedules *(Continued)*

| Schedule Name | Backup Backup Operation |
|---|---|
| Consolidate Quarterly | Completes a level 1 backup on the first day of each quarter (January, April, July, and October) and then consolidates that backup with the most recent full backup to produce the consolidated backup. The remaining backups for the quarter are incrementals. |

# Policies Resource

You use the Policies resource to create a lifecycle to use for both the *browse policy* and *retention policy* for your backed-up data. The client resource already has a default browse policy of Month and a default retention policy of Year assigned.

The browse policy determines how long the client file index maintains a browsable entry. If the browse policy has not expired, you can view the data in a graphical representation of the file system backed up, using the `nwrecover` program. After the browse policy expires, you can still use save set `recover` or the `scanner` program to recover the data, because save set information is still stored in the media database.

The retention policy determines how long the save set information is stored in the media database and how long the file remains retrievable from the backup volume. After all the retention policies for the save sets on a volume and other dependent save sets stored on other volumes expire, the volume is given a recyclable status and is available for reuse by Backup. Until the volume is relabeled, you can still use the `scanner` command to recover the expired save sets.

## Preconfigured Policies for Backup

Backup includes the preconfigured policies listed in the following table, which you can apply to either the browse policy or the retention policy.

**TABLE 2-5** Preconfigured Backup Policies

| Policy Name | Backup Behavior |
|---|---|
| Decade | Available 10 years |
| Month | Available 1 month |
| Quarter | Available for 3 months |
| Week | Available 1 week |

**TABLE 2-5** Preconfigured Backup Policies

| Policy Name | Backup Behavior |
|---|---|
| Year | Available 1 year |

For further details on how to manage save set life cycles, see “Specifying the Longevity of Backup Data” on page 111.

## Directives Resource

Directives contain instructions that can assist the backup process. For example, you can apply the UNIX With Compression directive to compress data from a UNIX client computer before it is sent to media during a backup.

### Preconfigured Directives Shipped With Backup

Backup ships with the preconfigured directives, as listed in the following table, that cover the most important and most useful backup instructions. You can use the UNIX standard directive as a template that you can customize for your UNIX clients. You can modify or delete most of the preconfigured directives, with the exception of the UNIX standard directive, which can be modified but not deleted. Note that Windows 98/95, OS/2, and Macintosh clients all use the DOS standard directive.

The Windows NT standard, DOS standard, and NetWare standard directives are shipped without specific directives coded, so you can customize them to your specific environment.

**TABLE 2-6** Preconfigured Directives

| Directive | Description |
|---|---|
| UNIX standard (Can be modified but not deleted) | Used for most of your UNIX clients, and when you do not need one of the other specialized directives. This selection:<br>• Applies the directive "+skip: core" to the *root* directory (/), thus skipping the backup of all *core* files.<br>• Applies the *swapasm* directive to the `/export/swap` directory to back up the relevant information about all NFS-based and local swap files, but not the data in them. If your swap files are located in a different directory, modify this line to include the appropriate location.<br>• Applies the *mailasm* directive to the `/usr/spool/mail` and `/usr/mail` directories to ensure that your mail files are backed up but not marked as read. If your mail files are located in different directories, modify these lines to include the appropriate locations.<br>• Applies the *logasm* directive to the `/nsr/logs, /var, /usr/adm,` and `/usr/spool` directories. If you have log files located in different directories, modify these lines to include the appropriate locations. |
| UNIX standard with compression | Used to back up and compress your UNIX clients. |
| DOS standard | Used to back up your DOS clients. |
| DOS standard with compression | Used to back up and compress your DOS clients. |
| NetWare standard | Used to back up your NetWare clients. |
| NetWare standard with compression | Used to back up and compress your NetWare clients. |
| Windows NT standard | Used to back up your Windows NT clients. |
| Windows NT standard with compression | Used to back up and compress your Windows NT clients. |
| Index | Used to back up the client file index. This option is usually used only by the Backup `savegrp` program. |

You can create your own directives to further increase the efficiency of client file backups. For further details, see "How to Create Customized Directives" on page 123.

# Notifications Resource

Backup provides several types of preconfigured notification messages that inform you about Backup activity: license status, client file index size, media attention, and the results of a scheduled backup.

Notifications are displayed in the Backup administration resource. They can also be sent to you through email or pager, or sent directly to a printer. The following table lists preconfigured notifications.

**TABLE 2-7** Preconfigured Notifications

| Notification | Backup Response |
|---|---|
| Bootstrap | Prints the bootstrap information on the Backup server's default printer. |
| Cleaning cartridge expired | Sends email to root: replace the expired cleaning cartridge. |
| Cleaning cartridge required | Sends email to root: insert a cleaning cartridge. |
| Device cleaned | Sends email to root: device cleaning is complete. |
| Device cleaning required | Sends email to root: clean the device. |
| Device disabled | Sends email to root: a device has been automatically disabled. |
| Index size | Sends email to root: check the size of the client file index because it will soon exceed the space available. |
| Log default | Directs the UNIX syslog facility (`/usr/bin/logger`) to log and distribute messages about all Backup events. |
| Migration attention | Sends email to root: check the status of a migration operation. |
| Migration completion | Sends email to root: a migration operation is complete. |
| Registration | Sends email to root: check the registration status. |
| Savegroup completion | Sends email to root: degree of success in completing scheduled backups, cloning, and archive operations. |
| SNMP notification request | Sends notifications through the configured trap. |
| Tape mount request 1<br>Tape mount request 2<br>Tape mount request 3 | Requests media be mounted in a device:<br>1. Displays a pending message.<br>2. Sends email.<br>3. Logs a message to `syslog`. |

### Backup Server Bootstrap Printout

When Backup finishes a scheduled backup that includes the Backup server, it creates a *bootstrap* save set, which contains the server index, media database, and Backup configuration files. The data in the bootstrap save set is the data you need to re-create your Backup server in case of a disaster. Backup writes the bootstrap file to media and automatically prints the bootstrap information to the default printer.

---

**Caution –** Save your bootstrap file in a safe location. The bootstrap information is essential for recovery from a disaster. For more information, see the *Solstice Backup 5.5 Disaster Recovery Guide*.

---

### Backup Administration Window Display

From the GUI version of the Backup administration program you can view progress messages and completion messages about the status of the backup.

- For progress messages, watch the Sessions display in the Backup administration window. The Sessions display information is also written to the `daemon.log` file in the `/nsr/logs` directory.
- For completion messages, watch the Messages display in the Backup administration resource. The Messages display information is also written to the `messages` file in the `/nsr/logs` directory.
- For information on why a backup cannot progress, view the messages in the Pending display. The Pending display shows requests to mount tapes and other conditions that require intervention.

The `nsrwatch` program also provides this information in a character-based format. For more information, see “Server Status Resource for Character-Based Displays” on page 295.

## Label Template Resource

Label templates provide a method to consistently name and label your backup volumes. The following table shows the preconfigured label templates that correspond with the preconfigured pools shipped with Backup. Backup automatically replaces *server-name* with your server's hostname. You can create a custom label template or let Backup create one for you when you create a custom pool.

**TABLE 2-8** Preconfigured Label Template Settings

| Pool Type | Preconfigured Setting |
|---|---|
| Archive | `server-name`.archive.001 |
| Archive clone | `server-name`_c.archive.001 |
| Default | `server-name`.001 |
| Default clone | `server-name`_c.001 |
| Full | full.001 |
| Migration | `server-name`.migrate.001 |
| Migration clone | `server-name`_c.migrate.001 |
| NonFull | NonFull.001 |
| Offsite | Offsite.001 |
| PC Archive | `server-name`.pcarchive.001 |
| PC Archive Clone | `server-name`_c.pcarchive.001 |
| Two Sided | `server-name`.001.a and `server-name`.001.b |

Use the Label Templates resource to create new label templates, which you can associate with a new pool or one that already exists. For further details, see "Customizing Label Templates" on page 83.

CHAPTER 3

# Server and Storage Node Operations

This chapter describes operations that you manage through the Backup server. This chapter consists of the following sections:

## Client/Server Communication Configuration

Communication between the Backup server and its clients is described by configuration values you enter in the Server resource and Clients resource. Backup relies on full and accurate configuration of the network to implement features that protect data and ensure security.

Each client resource should include both the DNS (domain name service) short name and long name in the Aliases attribute of the Clients resource in the Backup administration program.

For more details about how to set up Backup clients in the Clients resource, see "Configuring a New Backup Client" on page 102.

To diagnose problems with network communications that affect Backup, use the instructions in "Client/Server Communications" on page 377.

# Permissions Management

Although any user can view the server's resources from a client computer, only users you specify in the Administrator attribute in the Server resource can add to or change the configuration of the Backup server. When you first install Backup, `root@`*server-name* is the only user authorized to change the Backup configuration. To add other user IDs or computer names to the list of administrators, you must become root on the Backup server and add the other user IDs to the Administrator attribute in the Server resource.

Valid entries in the Administrator attribute include:

- *user@hostname*
- **@hostname*
- *user@**

(If you use the `nsradmin` interface to input these entries, you must separate them by commas.)

You can also add or restrict user privileges for individual clients. For more details, see "Allowing Remote Access Rights to Other Clients" on page 131.

---

**Caution –** When you add a user to the Administrator attribute in Backup, that person has Backup administration privileges for only that Backup server. Users in the Backup Administrator list do not automatically have root privileges on the Backup server or other computers in the network. A Backup administrator can change attributes for clients and other resources of the Backup server. Backup administrators have no special rights to client data for either backup or recovery.

---

# Firewall Support

Firewall support for Backup is a network-connection feature that supports *packet filtering* firewalls. When you use this feature, you do not have to open up large numbers of ports in the *firewall* to accommodate the Backup software.

With firewall support, you can determine how many ports are necessary for network connection. You can configure the Backup software to use a fixed range of ports for each system, whether it is a server, storage node, or client.

You configure network connections for two kinds of ports: the service port and the connection port.

## What Are Service Ports?

A *service port* is a listener port that provides services to clients. To avoid confusion, keep in mind that daemons are themselves clients of other services, so you should think about port configuration in terms of systems.

The default range used for service ports is 7937–9936, up to 2000 ports. The `nsrexecd` program, which must be running on the system, always listens on Port 7937. You do not need to include this number when you configure the service port range, but you should be aware that it is being used.

On the server, 11 services are registered, including the `nsrexecd` program. Ten of these services are required by the server. The additional service, the `nsrmmd` daemon, is required by each device. Therefore, 1 device requires 1 `nsrmmd` program, for a total of 11 services; 2 devices require 2 `nsrmmd` programs, for a total of 12 services; and so on.

## Configuring Service Ports

The port configurations are stored by `nsrexecd` on each system. Use the administration (`nwadmin`) or user GUIs (`nwbackup`) to change the configuration. You will need to restart your system after changing the configuration as changes cannot be applied until the system is booted.

The number of services is higher than the number of ports that you allocate for firewall support. Although the `nsrexecd` program uses a single port and the `nsrmmd` programs never share a port, other programs can share ports if they provide multiple services. For example, the sample `rpcinfo` output for Backup shown in Table 3-1 shows port allocations for 5 devices in addition to the 10 services required by the Backup server. A single port, 7937, is reserved for the `nsrexecd` program. Three ports, 7938–7940, are shared among 9 other services. Five ports, 7941–7945, are allocated for the 5 `nsrmmd` services. In total, 9 ports are allocated for 15 services.

**TABLE 3-1** Sample of `rpcinfo` output for Backup

| Program Number | Version Number | Protocol | Port | Program[1] |
|---|---|---|---|---|
| 390113 | 1 | tcp | 7937 | nsrexec |
| 390103 | 2 | tcp | 7938 | nsrd |
| 390109 | 2 | tcp | 7938 | nsrstat |
| 390110 | 1 | tcp | 7938 | nsrjb |
| 390103 | 2 | udp | 7939 | nsrd |
| 390109 | 2 | udp | 7939 | nsrstat |
| 390110 | 1 | udp | 7939 | nsrjb |
| 390107 | 4 | tcp | 7940 | nsrmmdbd |
| 390107 | 5 | tcp | 7940 | nsrmmdbd |
| 390105 | 5 | tcp | 7940 | nsrindexd |
| 390104 | 105 | tcp | 7941 | nsrmmd |
| 390104 | 205 | tcp | 7942 | nsrmmd |
| 390104 | 305 | tcp | 7943 | nsrmmd |
| 390104 | 405 | tcp | 7944 | nsrmmd |
| 390104 | 505 | tcp | 7945 | nsrmmd |

[1] Programs that register themselves to the portmapper do not have to use their program name as the service name. For example, `nsrexecd` registers its service as the `nsrexec` service as opposed to the `nsrexecd` service.

## What are Connection Ports?

A *connection port* is a port that is used to contact a service, whether it is on a server, storage node, or client.

The default range used for connection ports is 10001–30000, up to 20000 ports. The connection port ranges are stored in the `nsrexecd` client program. You determine the size of the range based on the type of Backup computers that are used (server, storage node, or client) and on what operations can be performed. See the following table for additional information.

# General Guidelines for Configuring Connection Ports

The default connection ports are as follows:

**TABLE 3-2** Default Connection Ports

| Type of Computer Allocating Connection Ports | Number of Connection Ports Required | Example |
|---|---|---|
| Client | Client Parallelism multiplied by 4 | Four clients require 16 connection ports. |
| Server with no storage nodes | Number of units of client parallelism multiplied by 2 | A server with 1 client requires 2 connection ports. A server with 5 clients requires 10 connection ports. |
| Server with storage nodes | The number of devices multiplied by 2, plus the number of autochangers multiplied by 2 | A server with 1 storage device and 1 autochanger requires 4 connection ports. A server with 5 storage devices and 5 autochangers requires 20 connection ports. |

## ▼ How to Configure Connection and Service Ports

The default ranges assigned are 10001-30000 for the connection ports and 7937-9936 for the service ports. You can change the default port configuration, as follows:

1. **As root, start the `nwadmin` program on the computer you want to configure ports for.**

   Note that you can view the port configurations for any host, but you can only apply changes to the ports from an `nwadmin` session run locally on the host.

2. **Select Configure Ports from the Option**

3. **In the Configure Ports dialog box, enter the client name (hostname or IP address) of the host and click OK.**

4. **In the Set Ports dialog box, enter a Service Ports range and a Connection Ports range, which is usually determined by the system administrator.**

5. **Click OK in the Set Ports dialog box to save your changes.**

6. **Click Cancel to exit the Configure Ports dialog box.**

**Caution –** You cannot restrict the number of ports necessary for network connections for clients that have pre-5.5 releases of the Backup software installed (without firewall support).

## Error Reporting

Server and storage node programs log port allocation failures to the log files. Client programs post a message as part of their `save` and `recover` operations, so manual operations get immediate feedback, and `savegrp` operations have error messages included in the completion notification.

# Storage Node Configuration

A *storage node* is a computer that is connected to a Backup server and to one or more devices used in the Backup server's backup, archive, and HSM operations. Devices attached to storage nodes are called *remote devices* because they are not physically attached to the controlling Backup server. The storage node runs special Backup software that controls devices. The data stored on media in remote devices is tracked in the media database and online client file indexes on the controlling Backup server.

Storage nodes you add to your Backup configuration can increase the Backup server's performance, give you more flexibility in designing your network, and centralize the control of data management activities to one or a few Backup servers.

To create a storage node, install the storage node binaries from the Backup software distribution on the storage node computer. Then define the storage node's devices. The method for defining devices is described in "Remote Device Configuration" on page 75.

The storage node hostname does not need to be on the server's Administrator list unless you run `jb_config` and `scanner` on the storage node. The entry for this is `root@`*storage_node_hostname*.

For an autochanger or silo, you must manually add the storage node's hostname to the Administrator list before you define the devices with the `jb_config` program. When the `jb_config` program is completed, you can remove the storage node's hostname from the Administrators list. If you need to configure a new autochanger later, you must add the hostname before you run the `jb_config` program again. After you add the storage node's hostname to the Administrator list, one instance of `nsrmmd` starts on the storage node for each device that it controls.

**Caution –** Do not attempt to convert an existing Backup *server* to a Backup storage node, as there is a potential for conflict when the resource database, media data base, and client file indexes from the retired server are merged into databases and client file indexes of the controlling Backup server.

## How to Configure a Timeout for Storage Node Operations

You can configure the amount of time that a Backup server should wait for a storage node request to complete through an attribute that is available in the Servers resource, named Nsrmmd Control Timeout. This attribute is available through the Details view in the `nwadmin` GUI, or through the Hidden option in `nsradmin`.

The Nsrmmd Control Timeout attribute controls how long the server's `nsrd` daemon should wait for a storage node request to complete. If the timeout value is reached without the completion of the request, the operation discontinues and an error message is logged.

The default value assigned to Nsrmmd Control Timeout is 5 minutes; you can specify any value within a range of 2 to 30 minutes.

In previous releases of the Backup software, this function was controlled through an environment variable, named NSR_MMDCONTROL. If the `nsrd` initializes on the server and detects that a setting for NSR_MMDCONTROL exists, the following informational message is issued:

```
“NSR_MMDCONTROL env variable is being ignored”
“use ‘nsrmmd control timeout’ attribute instead”
```

If you receive this message, follow these steps:

1. **Shut down the Backup daemons.**

2. **Remove the environment setting for NSR_MMDCONTROL.**

3. **Restart the Backup daemons.**

4. **Start the Backup administration program (either** `nwadmin` **or** `nsradmin`**).**

5. **Adjust the value of Nsrmmd Control Timeout to the setting that was previously assigned to the NSR_MMDCONTROL variable, or to one that best meets your current requirements.**

# Scheduled Backup Configuration

This section describes how to set up the scheduled backup features of Backup, including automated group backups and customizable backup schedules.

## Backup Group Configuration

Use backup groups to designate what time a client's scheduled backup starts. You can assign client save sets to backup groups to control which client's save sets are backed up at which times. You can also assign a client's save sets to more than one group.

If you have an especially large number of client computers, consider creating several groups with different start times to help reduce network traffic. For example, you could start the backup time of the group that includes the engineering department's computers at four o'clock in the morning, and the group with all other clients on the network at midnight.

If you create different groups, be sure to stagger their start times to avoid overloading the server. Schedule them far enough apart so that one group has completed its backup before the next group starts. Backup does not start a new group's backup until all the groups with earlier start times are finished. See "Example: Scheduling Large Client Filesystems" on page 44.

Backup provides several preconfigured groups for you to use. If you need a different group configuration, you can create new groups to fit your situation. To create and use a customized group, follow these steps:

1. **Create the group in the Groups resource.**
2. **Edit an existing pool or create a new pool in the Pools resource.**
3. **Select the new group in the Groups attribute.**
4. **In the Clients resource, edit or create the client resources for the client computers that contain the save sets you want to include in the group. Select the new group in the Groups attribute.**

---

**Caution –** Do not include spaces in a group name.

---

## How Backup Uses Backup Groups

The client save sets in each backup group begin their automatic scheduled backups according to the start time of the group. You can balance the backup loads by taking the client's backup schedule into account when you decide which clients to include in a specific group. (Refer to "Schedule Configuration" on page 41 for more information about creating schedules that vary the days that different clients perform full backups.)

FIGURE 3-1 on page 36 illustrates how Backup uses backup groups to back up multiple client save sets. In the example shown, three client computers—Oak, Elm, and Fir—are part of the group named Weekly Full, which starts its automatic scheduled backup at midnight. Client Oak runs a full backup of all its save sets every Monday and incremental backups of its save sets on the other days; client Elm runs a full backup of all its save sets on Tuesday and incremental backups on the other days; and client Fir runs a full backup of all its save sets on Wednesday and incremental backups on the other days of the week. Because each client runs its full backup on a different day of the week, the server is not overloaded.

The second group, "Accounting," illustrates how you can group clients by department. Group Accounting contains client computers Birch and Pine and starts its backups at 7:00 p.m., when the computers in the Accounting Department are available for backup. Although the two client computers run full backups on the same day, computer Pine is scheduled to back up only the `/usr/home` save set; all the save sets on computer Birch are backed up. By estimating how long a backup takes, you can determine what start time to set for the next group.

The save sets from each group are written to appropriate volumes mounted on storage devices. Backup uses *pools* to organize, track, and store save sets; it uses *groups* to determine what time clients start their scheduled backups.

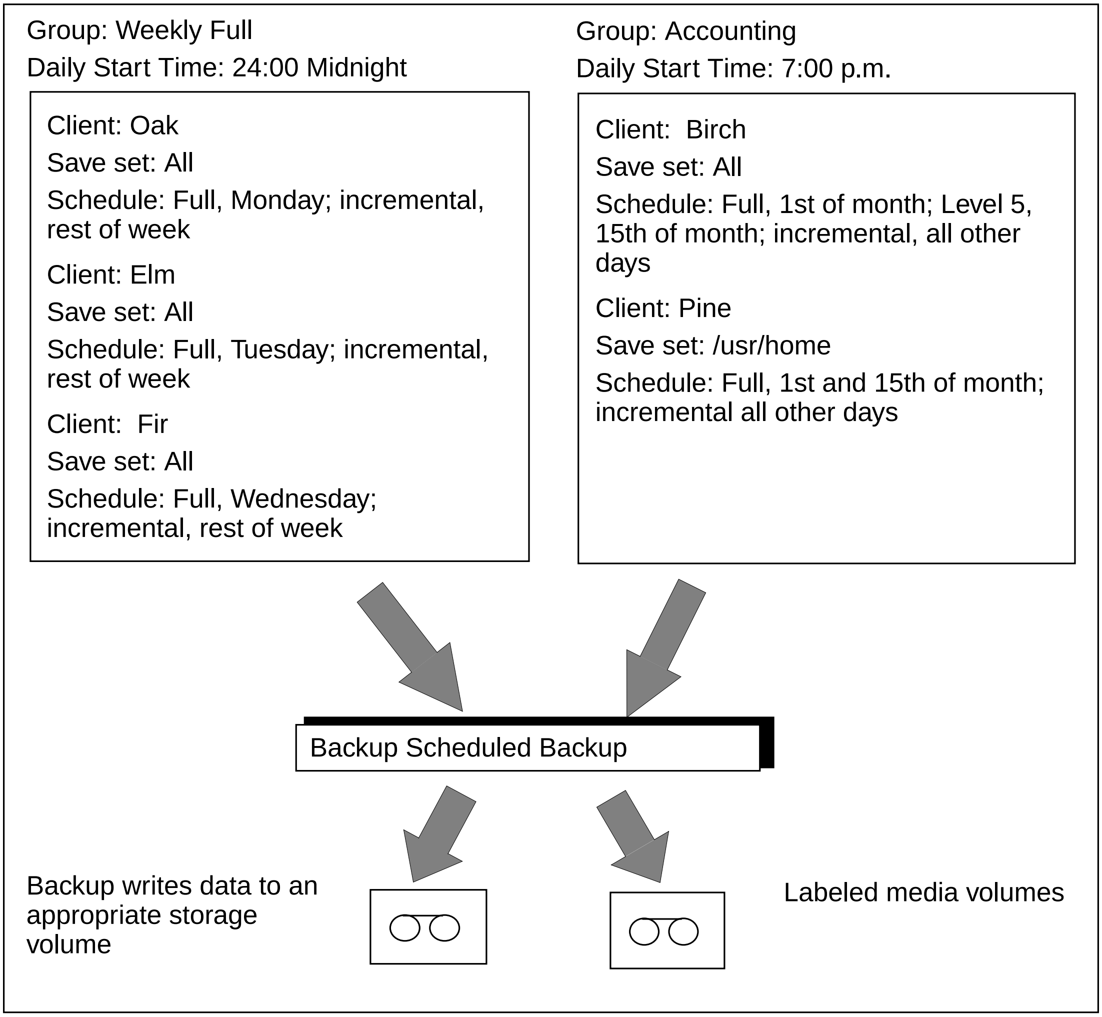

Group: Weekly Full
Daily Start Time: 24:00 Midnight
Client: Oak
Save set: All
Schedule: Full, Monday; incremental, rest of week
Client: Elm
Save set: All
Schedule: Full, Tuesday; incremental, rest of week
Client: Fir
Save set: All
Schedule: Full, Wednesday; incremental, rest of week
Group: Accounting
Daily Start Time: 7:00 p.m.
Client: Birch
Save set: All
Schedule: Full, 1st of month; Level 5, 15th of month; incremental, all other days
Client: Pine
Save set: /usr/home
Schedule: Full, 1st and 15th of month; incremental all other days
Backup Scheduled Backup
Backup writes data to an appropriate storage volume
Labeled media volumes

**FIGURE 3-1** How Backup Uses Groups to Back Up Multiple Clients

## Backup Default Group Settings

Backup ships with a preconfigured group named Default. To ensure that all data is backed up, Backup automatically adds all clients to the Default group. However, you must enable the default group for Backup to back it up. Depending on your needs, you can keep a client in the Default group, or you can put the client in one or more customized groups.

The two critical attributes in any group are the Start Time attribute and the Autostart attribute. The Start Time attribute for the Default group is set to start its daily backup at 3:33 a.m. You can change the Start Time attribute. You must enable the Autostart attribute for the Default group, and any other group you create, before Backup can run a scheduled backup of the group.

## Client Retries

If the Backup server cannot make a connection with a client, the Client Retries attribute in the Groups resource specifies the number of times that the server should try to connect to the client before the effort should be considered a failure. The first retry does not occur until after an attempt has been made to contact each client (at a minimum). The Inactivity Timeout attribute in the Groups resource specifies the number of minutes that the Backup server waits for evidence of backup activity on the client. If the server has not received status information for longer than the time specified, the server abandons the backup operation for the save set.

The backup of an abandoned save set might be completed, but the automated report from `savegrp` will not show that the backup is completed. For example, if the client is being backed up over Sun's distributed computing file system (NFS™) connection and the NFS server crashes and reboots, the Backup backup hangs until it times out. The Backup server marks the save set "abandoned," and when the NFS system server comes back up, the backup continues and is completed.

The preconfigured attributes for the Default group are described in "Groups Resource" on page 18. You can make changes to any Default group attribute, but you cannot delete the group. You can, however, create or delete as many customized groups as you need.

## Using a Group Schedule to Override a Client's Regular Backup Schedule

You can use a group's Level and Schedule attributes to override a client's regular backup schedule. For example, one evening you might want to run a full backup on all the clients in a group, regardless of the clients' regular backup schedules. The entry you make in the Level attribute overrides the backup level setting for every client in the group.

Alternatively, you might want a group of clients to follow the same backup schedule instead of each client's individual schedule. You could assign a group of clients to follow the default schedule (full every Sunday) regardless of each client's individual schedule. If you leave the group's Level and Schedule attributes blank (the preconfigured setting), the clients follow their individual backup schedules.

## Monitoring and Managing Group Backups

Use the Group Control window in the Backup administration program to monitor scheduled groups during a backup. The Group Control feature, savegroup completion message, bootstrap printout, and system console log provide information about the success of scheduled backups and the information you need to recover your data.

The Group Control feature provides status information and contains controls for previewing, stopping, and starting scheduled backup groups.

The status information about the most recently started backup group is displayed as one of the following:

- Running
- Never Run
- Finished
- Not Finished (indicates the backup has exited without finishing)
- Preview Run (indicates the test of the backup configuration)

### Monitoring Operations

The Group Control Details feature available in the Backup administration program enables you to view more detailed information about a completed group backup. Use this feature to determine which client save sets were backed up successfully and which save sets failed.

The Group Control Details window displays the status of client save sets in the backup process in one of three message fields:

- Pending Save Sets – displays the client save sets that have not yet been backed up
- Completed Save Sets – displays the client save sets that Backup has successfully backed up
- Failed Save Sets – displays the client save sets that Backup did not back up (typically because of a computer or network crash)

You can use the Group Control Preview feature to simulate a backup for a specific group. This feature helps you identify potential problems before Backup runs an upcoming group backup. To preview a backup group with the Backup administration program, display the Group Control window and click the Preview button. To preview a backup group from the command line, become root on the Backup server, then issue the `savegrp -p` *group-name* command at the shell prompt.

Backup displays information about how a group will perform during its next scheduled backup, instead of displaying past information about completed group backups.

## Immediate Start of a Scheduled Group Backup

When you start a scheduled backup group manually (on demand), Backup runs the backup at the level of the next scheduled backup, which can be full, level 1-9, consolidated, or incremental.

To start a group backup immediately, complete one of the following procedures:

- In the Backup administration program, click the Start Now button in the Group Control window.
- From the command line, become root on the Backup server, then issue the `savegrp` *group-name* command at the shell prompt.

When you use the Start Now control, Backup overrides the Groups scheduled start time and immediately backs up the clients in the group.

## Stop and Restart of a Scheduled Group Backup

After you initiate a Stop in the Group Control window, Backup completes its backup of the current save set, halts the rest of the scheduled backup, and displays Not Finished in the Status field in the Group Control window.

After you initiate a Restart through the Group Control window, Backup resumes the scheduled backup for the group and displays Running in the Status field.

# Management of Open Files During a Scheduled Backup

If a client’s open files change during a scheduled backup, Backup backs up the old version of the files and detects that they are changing. A warning message similar to the following appears in the Group Control Details window:

```
warning: file filename changed during save
```

The changes to the file are not backed up. To back up the changes, you can restart the backup group or allow Backup to back up the client during the next scheduled backup.

## Savegroup Completion Message

When the backup is completed, Backup generates a report about the success of the scheduled backup. Backup sends the root user an automatic notification and displays the same information in the Backup administration program.

## Bootstrap Generation and Printout

When the backup group includes the Backup server, Backup generates a special save set called the *bootstrap*, which includes the server file index, media database, and configuration files. *The bootstrap information is essential for recovery from a disaster.* Refer to the *Solstice Backup Disaster Recovery Guide* for information on how the bootstrap is used during a disaster recovery operation.

By default, the bootstrap is printed to the Backup server's default printer. To change the default printer, change the Printer attribute in the Groups resource.

A bootstrap printout is created with any scheduled backup of a group that includes the server, or after other scheduled backups if the server is not in an active group. A bootstrap printout is generated whether the scheduled backup is initiated automatically or manually.

You can save the bootstrap to a file or email it to one or more user ID. To save the bootstrap to a file, run `nwadmin` and select Notifications from the Customize menu, and then select Bootstrap. The Action attribute displays:

```
# /usr/bin/lp -s -c -t bootstrap -d%PRINTER
```

Change this to:

```
# /bin/cat >> /directory/filename
```

To e-mail the bootstrap file to more than one user ID, change the line to:

```
# /usr/ucb/Mail -s “nwserver bootstrap” user-name@corp.com
```

## System Console Log

The UNIX system log displays messages passed from Backup. When Backup is installed, it adds lines to the configuration log file (`syslog.conf`) to tell the system log facility what types of notices to direct to which file or user. For example:

```
daemon.notice                    /dev/console
daemon.notice                    /nsr/logs/messages
daemon.notice                    operator
local0.notice                    /nsr/logs/summary
local0.alert                     root, operator
```

## Schedule Configuration

The Backup server determines the amount of data to back up for each client system across your network according to the backup schedule you assigned to each client. Schedules can be very simple or very sophisticated, depending on the needs of your environment. All clients can share the same schedule, or each client can have its own schedule. Use the Schedules resource to create customized schedules that you can apply to client save sets through the Clients resource. See Chapter 5, "Client Operations," on page 101 for more information about the Clients resource and client configuration.

### How Backup Uses Backup Schedules

Backup uses a client's backup schedule to determine what level of backup operation to perform on a given day for the specified save sets. The time of day the backup operation begins is determined by the Start Time assigned to the Groups resource with which the client save sets are associated.

Backup supports five different types of backup levels:

- Full – backs up all files, regardless of whether they have changed since the last backup operation.
- Level 1-9 – backs up files that have changed since the last *lower numbered* backup level.
- Incremental – backs up files that have changed since the last backup, regardless of the level.
- Consolidated – backs up all data that has changed since the last full backup and subsequently merges these changes with the last full backup. This creates a new full backup.
- Skip – skips the scheduled backup.

(See "Backup Levels" on page 45 for a detailed description of backup levels.)

Use the Schedules resource to customize backup schedules to best suit your needs. For example, some clients may have data you want to back up at level "full" every three days, with incremental backups in between. Other clients may have less critical data that only needs a full backup once a month, with incremental backups or level 1-9 backups on other days.

You can use backup schedules to balance and stagger the load on your Backup server. Depending on the size of your network, you can apply the same schedule to all clients. For example, if no one works on Sunday and you want to run full backups on that day, you can apply the default schedule to all your clients. The default schedule tells Backup to perform full backups on Sunday and incremental backups the rest of the week. The following figure illustrates how the default schedule works for three clients: Client A, Client B, and Client C.

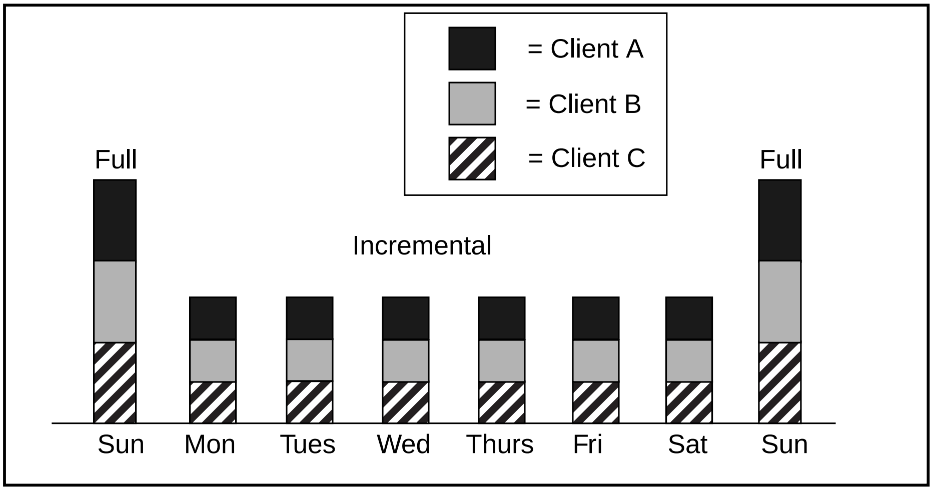


**FIGURE 3-2** Using the Backup Default Schedule for Multiple Clients

Because full backups can take a long time, you may want to stagger them throughout the week. For example, you can apply a schedule that performs a full backup for Client A on Sunday, a second schedule that performs a full backup for Client B on Tuesday, and a third schedule that performs a full backup for Client C on Thursday. The following figure illustrates how you can use staggered backup schedules for multiple clients.

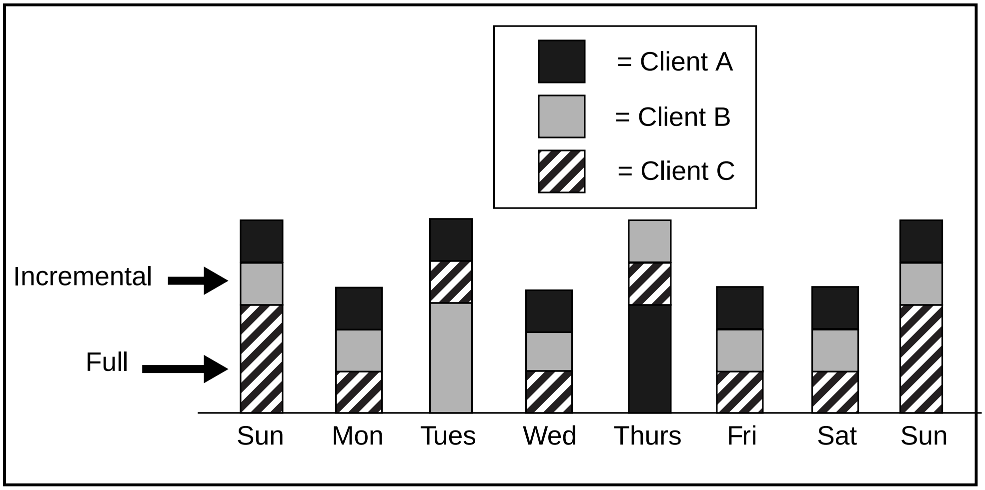


**FIGURE 3-3** Staggered Weekly Schedules for Multiple Clients

When you balance and stagger the load on your Backup server, you can increase server efficiency. Using different start times for groups of clients also helps increase server efficiency.

## Backup Schedules

Backup makes it easy to set up your backup schedules. Deciding which backup schedules best fit your environment, however, requires a planned strategy.

When you create backup schedules, consider the following factors:

- The amount of data you have to back up
- The number of volumes you want to use
- The time available to complete a backup
- Whether the number of volumes required to recover from a disaster, such as a disk crash, matter

Additionally, you must determine a policy for recovering files. For example, if users expect to recover any version of a lost file for at least three months (that is, the retention policy is equal to three months), you need to maintain all the save set entries in the media database for three months. On the other hand, if users only expect to recover data from the last month, you can use level 1-9 backups to decrease the quantity of volumes you need to maintain.

The length of time that data is available for Backup to recover is determined by the browse and retention policies associated with each client. See "How the Browse and Retention Policies Manage the Data Life Cycle" on page 116 for more information about how Backup manages the data life cycle.

## Example: Scheduling Large Client Filesystems

At a moderate backup rate of 400KB per second, a full backup for a client with 10GB of data takes about 5.5 hours to complete. Consequently, it may not be convenient to complete a scheduled, full backup for client save sets as large as this because of the amount of time the backup takes.

You can schedule the client's disk volumes for backup at different times by separating them into different backup groups. When you split one client's save sets into multiple backup groups, you back up all the client's files, but not all at once. It is less time-consuming than a full backup of all the local data at one time.

To back up the client's filesystems individually, add and configure the same client several times addressing the different filesystems in the Clients resource. For example, configure the first client resource to back up one filesystem, `/usr`, with one backup schedule in one group, and configure the second client resource to back up another filesystem, `/var`, with a second backup schedule in another group.

---

**Caution –** When you create separate backup schedules and explicitly list save sets, any files or file systems not included in an explicit list are omitted from backup. This includes any new disk volumes that are added to the system. This risk of omission does not exist when you enter the special value "All" in the Save Set attribute; Backup automatically adds the new disk volumes to the backups.

---

## Schedule Configuration Attributes

To create a customized backup schedule, you must define the following schedule configuration values in the Schedule resource:

- Name

Choose a simple, descriptive name, for example "Monday Full." You cannot change the Name attribute of an existing Schedule resource. For example, if you want to change the schedule "Full Every Friday" to "Full Every Monday," you must create a new "Full Every Monday" schedule. You cannot change the existing schedule to complete full backups on Mondays instead of Fridays, and then edit its name.

- Period

Specify how often you want a backup to run. When you select Week, the backup level is applied to that day of the week for all the weeks in the calendar year, for example, full backups every Sunday. When you select Month, the backup level is applied to that day of the month for all months in the calendar year, for example, full backups on the fifteenth of each month. Week is the default setting.

- Level

Select the backup level for each day in the period. Valid values for backup level include “full,” “incr,” “consolidated,” and “1-9.” See *Backup Levels* below for more information on backup levels.

- Override

Specify a list of actions and dates overriding an existing backup level for a specific day. For example, you may not want a full backup to run on a holiday. You can override the schedule so the full backup runs on the day before or the day after the holiday.

You might also want to change the Force incremental setting which is located in the Groups resource. The default setting for this attribute is “Yes.” This means an incremental backup will occur if the group is run more than once a day. Set this attribute to “No” to do more than one full backup per day.

## Configuration Order for Backup Schedules

If you want to use your own customized schedule, you must configure the schedule before you can associate it with a client or save set in the Clients resource. The start time for your automatic daily scheduled backup is determined by the backup group with which the client save sets are associated. The length of time that the data is available for browsing or recovery is determined by the browse and retention policies you configure for the client’s save sets, rather than by the schedule.

# Backup Levels

Because it might not be practical or efficient for you to run a level full backup every day, Backup enables you to specify the level of the backup operation performed during its automatic, scheduled group backups. Limiting how often you perform a full backup can help maintain server efficiency, while still ensuring that your data is protected. Different backup levels enable you to trade off the number of volumes and amount of time required to complete a backup with the number of volumes and amount of time required to recover from a disk crash.

Backup supports five kinds of backup levels for filesystem data:

- Full – backs up all files, regardless of whether they have changed.

- Level 1-9 – backs up files that have changed since the last *lower numbered* backup level, the last full backup being considered a level zero. For example, a level 1 backs up all the files that have changed since the last full backup (considered a level 0). A level 3 backs up all the files that have changed since the last level 2, level 1, or full backup. A level 9 backs up all the files that have changed since the last level 8, 7, 6, 5, 4, 3, 2, 1, or full backup.
- Incremental – backs up files that have changed since the last backup, regardless of the level.
- Consolidated – performs a level 1 backup and subsequently merges these changes with the last full backup. This backup level is the same as a full level backup, however, without incurring the same heavy network traffic.

  When you initiate a consolidation backup, Backup performs a level 1 backup, provided there has been a preceding full backup, and then merges the new level 1 with the preceding full backup. All subsequent backups are consolidation backups. The consolidation process requires that you use two media drives simultaneously; one for reading the recently backed up data and one for writing the recently consolidated data.
- Skip – skips the scheduled backup. For example, you may want to skip a backup on a holiday if you know that no one is available to change or add more volumes.

---

**Caution –** A level Skip backup does not back up any data, however, the Backup server still contacts the client for the scheduled backup at the Start Time indicated for the backup group. The server's `savegrp` program generates a *Savegroup Completion Report* that shows that a level Skip backup was performed and no data was backed up. Any problem that could cause the Backup server to hang waiting for the client to respond has the same effect during a level Skip backup as for other level backups, even though no data is actually backed up.

---

## How Backup Uses Backup Levels

A backup schedule defines what level backup should be performed on a given day during a backup cycle. You can apply one or more of these backup levels to customize a backup schedule. If you are considering using backup levels in a customized schedule, consider the following issues to help you make decisions that best suit your environment:

- Full backups take more time to complete than incremental backups.
- If you have only one storage device and the full backup does not fit on a single piece of media, an operator must be available to monitor the backup and change the media.
- Full backups cause the online indexes to grow more rapidly than incremental or level backups.

- Level backups serve as checkpoints in your schedules because they collect all the files that have changed over several days, or even weeks, into a single backup session. Using level backups can simplify and speed file recovery.
- Consolidated backups provide the same benefits at the same cost as do full backups. The essential difference, however, is that consolidated backups are less taxing on the network and client because only a level 1 backup is performed. The server's performance, however, is slower because the server consolidates the changed data with the original backup.

**TABLE 3-3** Advantages and Disadvantages of Backup Levels

| Backup Level | Advantages | Disadvantages |
|---|---|---|
| Full | • Faster restore | • Slow backup<br>• High server load<br>• High load on client and network<br>• Uses more volume space |
| Level | • Faster backup<br>• Low load on server<br>• Uses least volume space | • Slow restore<br>• Data can spread across multiple volumes |
| Consolidated | • Faster backup (from the client's perspective only)<br>• Faster restore<br>• Low load on client and network | • Longest high load on server<br>• Requires at least two volume drives<br>• Uses most volume space |

**Caution –** The online client file indexes, server index, and media database are backed up whenever the Backup server is backed up. In general, they are backed up at the same level as the server. For example, if the Backup server's backup is a level full, the backup of the online client file indexes, server index, and media database is also a full; if the Backup server's backup is a level 5, the backup of the online client file indexes, server index, and media database is also a level 5. However, *when the server's backup level is incremental, the backup of the online client file indexes, server index, and media database is level 9*. For the consolidated backup, the backup of the online client file indexes, server index, and media database is a full level backup. It does not perform a level 1 backup for this data.

## How Backup Levels Work

Backup levels work in conjunction with a client's backup schedule. The way you define the backup levels directly affects how long the recovery from a disk crash takes and how many backup volumes you need.

The following paragraphs, and the following figures, illustrate the concept of how backup levels work and the data requirements for recovery in the event of data loss.

On October 2, a full backup runs. On October 3, the incremental backup saves everything that changed since the full backup. On October 4, the incremental backup backs up everything that changed since the 3rd. On October 5, the level 7 backup backs up everything that changed since the full backup. To fully recover from a disk crash on October 5, you need only two volumes: the full volume and the level 7 volume. You no longer need the data on the volumes from October 3 and 4, because the level 7 volume includes that information. (See the following figure.)

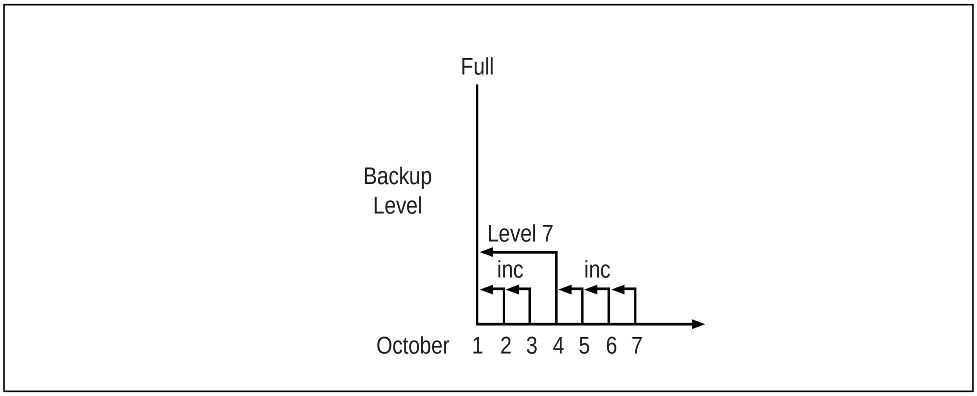


**FIGURE 3-4** Example: Backups for October 2 Through October 8

On October 6, 7, and 8, the incremental backup backs up everything that has changed since the level 7 backup. On October 9, as shown in the following figure, the level 5 backup backs up everything that changed since the full backup. To fully recover from a disk crash on October 9, you need only two volumes: the full volume and the level 5 volume. You no longer need the data on the volume from the level 7 backup or the subsequent incremental backups because the level 5 volume includes that information.

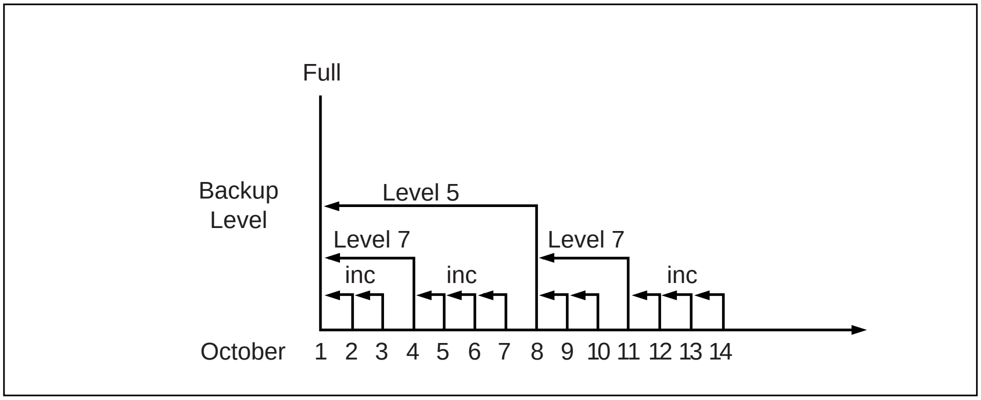


**FIGURE 3-5** Example: Backups for October 2 Through October 15

On October 12, the level 7 backup backs up all the data that changed since the last lower numbered backup, in this case the level 5 backup from October 9. To recover from a disk crash on October 12, you need three volumes: the full volume, the level 5 volume, and the new level 7 volume. (See the previous figure.)

On October 16, the level 5 backup backs up all the data that changed since the last lower numbered backup. Because no lower numbered level backup has been performed (for example, levels 1–4), the level 5 backup backs up all the data that changed since the full backup. To recover from a disk crash on October 16, you need two volumes: the full volume and the new level 5 volume. (See the following figure.)

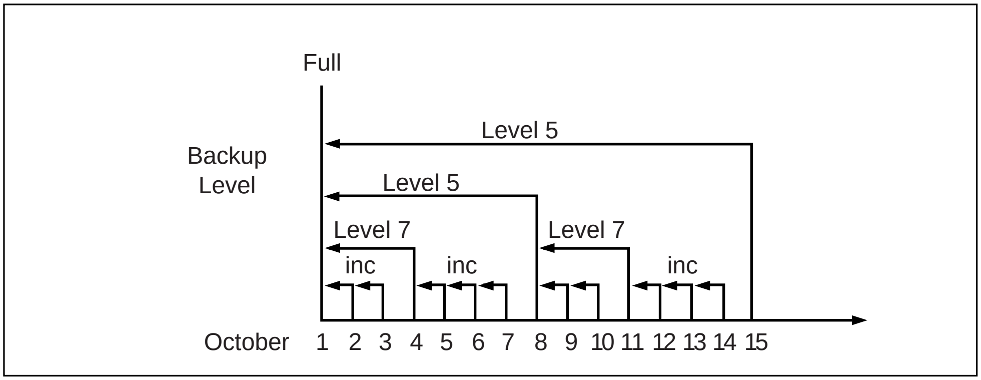


**FIGURE 3-6** Example: Backups for October 2 Through October 16

On October 17, the consolidated level “c” automatically performs a level 1 backup; this backs up all the data that has changed since the last full back up. As part of its consolidation process, the Backup server merges this level 1 backup with the last full back up (created on October 2) and builds a new full level backup. To recover from a disk crash on October 17, you need one volume, the full volume that was created using the consolidation process on October 17 (see the following figure).

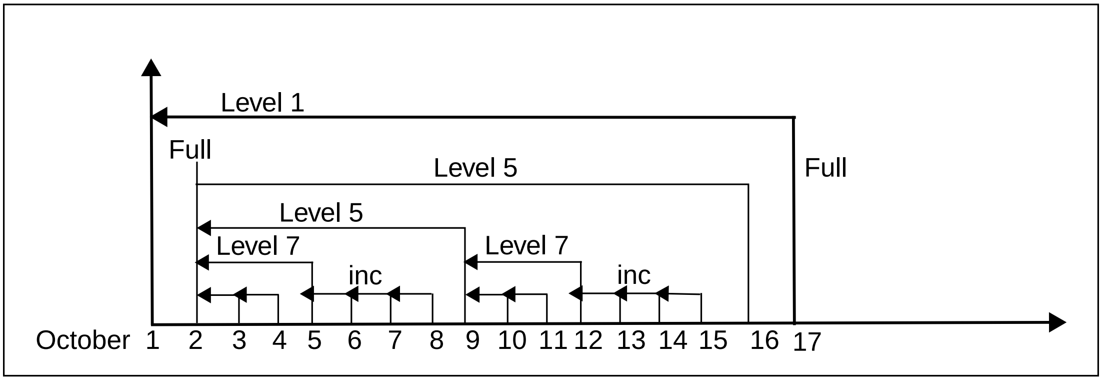


**FIGURE 3-7** Example of a Consolidated Backup

Level 1-9 backups help you maintain control over the number of volumes you use. A carefully planned backup strategy enables you to recover everything to disk with a minimum number of volumes. The fewer volumes you need to recover from a disk crash, the less time you must spend restoring the disk.

You can also control the size and time it takes to back up your data by using directives, which compress and eliminate unnecessary data from your backups. For example, you can use a directive that tells Backup to skip certain files or file systems when performing a backup. For more information, see “What Are Directives?” on page 123.

# Save Set Consolidation

This section describes the save set consolidation (SSC) feature. Save set consolidation merges a new level 1 backup with the last full backup of a save set to create a new full backup.

Essentially, save set consolidation is a process rather than an attribute; it describes the process of merging a level 1 backup with an existing full level backup. Consolidation is really a full level backup and the resulting save set of the consolidation process is the same as a full level save set. Although a consolidated backup takes place, there is really no such thing as a consolidated save set.

You cannot find “consolidation” as a backup level in most of the various resources where backup levels traditionally appear. For example, you cannot find “consolidation” as a potential backup selection under Pool in the Media menu in the nwadmin Graphical User Interface. The one exception is the Schedules attribute, where you can use the default Consolidation schedules provided or create your own consolidation schedule by overriding a scheduled level with the special level “c.”

Save set consolidation eliminates the need to perform full backups at regular intervals. After scheduling a save set consolidation backup, you only perform one full backup during the first scheduled consolidated backup cycle. Afterward, all subsequent backups of the consolidated save set are incremental backups.

Save set consolidation takes place entirely on the server. The burden of backups is shifted entirely to the server, reducing client resource use and network traffic. This shift to the server also results in more frequent level saves, resulting in fewer tapes to process in the event of a full restore.

When working with large files, save set consolidation shortens the backup window by doing incremental saves, thus reducing the number of tape drives required during a recover.

---

**Caution –** If there are no existing level Full backups the first time a consolidation backup begins, the consolidation backup defaults to a level Full. Once a level Full backup exists, subsequent consolidation backups are performed as a level 1 followed by consolidation with the existing level Full.

---

There are some limitations to the save set consolidation process:

- Renamed directories cannot be consolidated.
- Deleted directories are not supported for non-UNIX clients.
- You can only administer the save set consolidation server with a nwadmin program from a release that includes the save set consolidation feature.

When these limitations exist, save set consolidation fails with an error message.

---

**Caution –** Even if a consolidated backup cannot be completed because of a system crash, tape drive errors, or other problem, data integrity is ensured. The consolidated backup will be aborted. The transactional log assists save set consolidation in putting back changes made to the online index. In most instances, Backup only makes online index changes when data has been written to tape.

---

## ▼ How to Direct Data From a Consolidation Backup to a Specific Pool

By default, save sets from a consolidation backup are written to whatever media is mounted for the group most recently backed up. If you want to direct consolidated save sets to a specific set of media, follow these steps:

1. **Configure a Group resource for consolidation backups.**
2. **Configure a Pool resource for consolidation backups, and select the name of the Group resource you created in Step 1 as the choice for the Groups attribute.**
3. **Edit the Client resource for each client for which you want to perform consolidated backups, and assign the client to the Group that you created for consolidation backups.**

## Hardware Requirements

Save set consolidation requires at least two attached tape drives. For better performance, you should have three or more tape drives available for consolidated backups.

# Consolidated Backup

The following example and steps explain how save set consolidation works:

1. You must set up the schedule for consolidation backup. For example, on a weekly schedule, the first day would be designated as the consolidated full backup, with the remaining days of the week being designated as incremental:

| Mon | Tues | Wed | Thu | Fri | Sat | Sun |
|---|---|---|---|---|---|---|
| Cons | Incr | Incr | Incr | Incr | Incr | Incr |

The first backup of any new save set  is regular full backup and not a consolidated full backup.

2. During the second week, a real consolidated full backup is scheduled. The Backup server, however, gives priority to any other regular backup and restore jobs. When these jobs are completed, the consolidated backup is invoked and completed.

3. For the consolidated backup process, the Backup server initially requests that the client perform a level 1 backup. This backup encompasses the file changes between the current time and the last full backup save time.
4. Once the initial level 1 backup is completed, the consolidation backup process browses the index entries for changed files in level 1 entries. In the process, it builds a list of changed files (new, modified and deleted) along with their corresponding save set/volume information. This list is used for determining which files are to be extracted and which volumes are to be mounted for the consolidation process.

When the list is built, the Backup server allocates two tape drives for the consolidated backup operation. Once allocated, the server mounts the tape with the last full backup and the destination tape. The server then processes all files on the list in the following manner:

- If a file has not been updated since the last full backup, it is then extracted from the last full tape and stored in the destination tape.
- If a file has been changed since the last full backup, then the appropriate volume with the given file is mounted. The file is then extracted and backed up.
- The catalog is changed to reflect the addition of the new consolidated full backup.

# Using Save Set Consolidation

You can invoke save set consolidation by using the GUI or one of two commands on the command line.

## ▼ How to Invoke Save Set Consolidation Through the GUI

1. **Select the Customize option.**
2. **Select Schedule.**
3. **Click on the day during which you want the save set consolidation to occur. Select consolidate from the levels menu that appears. The letter “c” signifies this newly created consolidated backup.**

Just as with other backup levels, Backup will consequently invoke the save set consolidation backup on the scheduled date.

For a consolidated backup, Backup will first perform a level 1 backup; subsequently, it performs a consolidation of the level 1 backup and the most recent full save set.

## Invoking Saveset Consolidation from a Command Line

You can invoke save set consolidation through two different programs:

- savegrp(8)

  The `savegrp` program must be used with the `-lc` option (to indicate that the backup level is "c"), or with a schedule that has a level "c" on the schedule. This program automatically performs a level 1 backup, followed by the actual consolidation process.

- nsrssc(8)

  The `nsrssc` program completes the consolidation process. For the `nsrssc` program to be successful, a level 1 save set and a level full save set must already exist.

Both programs also offer other options to maximize the flexibility of save set consolidation. For further information, refer to the `savegrp(8)` and `nsrssc(8)` man pages.

# When to Use Save Set Consolidation

You should use save set consolidation if the following conditions are present:

1. **A client is at a remote location and data transfer over the network to the server is a performance issue for either the network or the client.**

2. **Either your network bandwidth is small, or large backups over the network are cost-prohibitive.**

3. **You need to back up large file systems with little incremental data.**

4. **The server has the necessary resources (a minimum of two volume drives and preferably three volume drives) and the workload capacity to consolidate full backups locally.**

*Do not* use save set consolidation if any of the following conditions are present:

1. The client is connected to the server over a fast network or is a local client, and the network data traffic generated by full backups is not a problem. In this instance, save set consolidation will not produce a measurable benefit.
2. The file systems being backed up are not very large or contain a large number of small files that are modified often.
3. The incremental data usually contains a large amount of data, and the number of files changed since the last full backup is large compared to the total number of files in the file system.
4. It is cost-prohibitive to allocate three (or the minimum of two) tape drives for the exclusive use of the server while it consolidates the full backup.

# Index Management

Backup tracks the files it backs up in the client file indexes and the media database. The client file indexes keep track of the files that belong to a save set, and the media database tracks the name of the volume, the backup dates of the save sets on the volume, and the filesystems in each save set. Backup can automatically control the size of the client file indexes and media database according to the browse policies and retention policies you set. For more details about using browse and retention policies, see "How the Browse and Retention Policies Manage the Data Life Cycle" on page 116.

## Client File Index Format

The client file index is stored on the Backup server as one UNIX directory for each client computer, in the format `/nsr/index/`*clientname*. A volume header file appears in every directory and functions as a directory structure for the files contained within. Within the directory are an unlimited number of files.

The file names within the client file index directory include several different *.suffix* endings to identify the file types. Table 3-4 lists the files that can appear in the /nsr/index directory maintained for each individual client.

**TABLE 3-4** Files Contained in Each Individual Client File Index

| Filename | Function |
|---|---|
| VolHdr | Volume header for the directory. This file contains a list of the segments that have been allocated, the file descriptors, and other accounting and control information needed by nsrim and nsrck. |
| sr | Bitmap of the logical sr data file, which is the file that contains the actual data for each file entry in the client file index. |
| sr.0 | First segment of the logical sr data file. One saverec (about 220 bytes) is created for each file that participates in a backup. |
| sr.1 through<br>sr.*n* | Remaining segments of the logical sr data file. |
| sr_i0 | Bitmap of the logical sr_i0 file, which contains the hashed keys (B-tree) for the data by filename. |
| sr_i0.0 | First segment of the logical sr_i0 index file. |
| sr_i0.1 through<br>sr_i0.*n* | Remaining segments of the logical sr_.i0 data file. |
| sr_i1 | Bitmap of the logical sr_i1 file, which contains the hashed keys (B-tree) for the data by internal file ID number. |
| sr_i1.0 | First segment of the logical sr_i1 index file. |
| sr_i1.1 through<br>sr_i1.*n* | Remaining segments of the logical sr_i1 data file. |
| *filename.x* | Contains a list of records to be deleted. These files function as transaction logs and ensure consistency in the event that a save set backup is interrupted (for example, due to system crash). |
| *filename*.t | A temporary file used during sorting. |

# Media Database Format

The media database is also stored on the Backup server, and follows the same structure as the client file index. Filenames within the directory /nsr/mm/mmvolume follow the same naming convention as is evident in the client file index structure. The example that follows contains the nsrls -f output and the ls -l output generated when these commands are executed on /nsr/mm/mmvolume:

```
jupiter# nsrls -f /nsr/mm/mmvolume
Database id 0: mm/mmvolume
 Fid |      Size |     Count | Name
------------------------------------------
   0 |     16 KB |         2 | vol
   1 |     16 KB |         3 | ss
   2 |     16 KB |         2 | vol_i0
   3 |     16 KB |         1 | vol_i1
   4 |     16 KB |         1 | vol_i2
   5 |     16 KB |         1 | vol_i3
   6 |     16 KB |         0 | vol_i4
   7 |     16 KB |         3 | ss_i0
   8 |     16 KB |         3 | ss_i1
   9 |     16 KB |         0 | ss_i2
  10 |     16 KB |         3 | ss_i3
jupiter# ls /nsr/mm/mmvolume
VolHdr    ss_i2.0   vol_i1.0
ss        ss_i3     vol_i2
ss.0      ss_i3.0   vol_i2.0
ss_i0     vol       vol_i3
ss_i0.0   vol.0     vol_i3.0
ss_i1     vol_i0    vol_i4
ss_i1.0   vol_i0.0  vol_i4.0
ss_i2     vol_i1
jupiter#
```

The mmvolume directory can also contain a file /nsr/mm/.cmprssd. This file functions as a timestamp to track media database compression. The .cmprssd file ages automatically and after 22 days, the media database is compressed. To force immediate compression of the media database, shut down the Backup services (with the nsr_shutdown command), remove the .cmprssd file, and restart Backup.

## Index Size and Structure

The structure of the client file indexes avoids operating system restrictions on file size and allows the client file index for a single client to continue to grow. As the client file index grows, it splits into segments of 2 GB each. If you want to check the size of a client's file index, enter the `nsrls -f` command, as in the following:

```
# nsrls -f /nsr/index/client-name/db
```

The path in the example is the default path. To change the path where the index resides, change the value in the Index Path attribute in the Details view of the Clients resource.

The UNIX `ls` command will not provide an accurate report on the size of the client's file index. To obtain an accurate report on the size of a client's file index, use the `nsrls` command. The output of the `nsrls -f` command is a table, for example:

```
# nsrls -f /nsr/index/mars/db
Database id 0: /nsr/index/mars/db
Fid |      Size |     Count | Name
------------------------------------------
   0 |     16 KB |        15 | sr
   1 |     16 KB |        15 | sr_i0
   2 |     16 KB |        14 | sr_i1
```

## ▼ How to Manually Reduce the Client File Indexes and Media Database

You can also use manual methods to control the size of the client file indexes and the media database:

- Purge

```
# nsrmm -d -P volume-name
```

This method removes all relevant entries for user files on that volume from the appropriate client file indexes, but retains the volume in the media database. Purging a volume does not destroy the contents of the tape. You can still recover the contents using the `scanner` program.

- Delete

```
# nsrmm -d volume-name
```

  This method removes the volume’s entry from the media database. It also removes all the relevant entries for the user files on that volume from the client file index. Deleting a volume does not destroy the contents of the tape. You can still recover the contents using the `scanner` program.

- Recycle

```
# nsrmm -m -R volumename
```

  This method relabels the volume, deletes the volume from the media database, and reinitializes the tape. After a tape is recycled, you cannot use the `scanner` program to recover the contents.

You must wait until the purge operation is completed before running the `nsrck` program or performing a recovery. The purge or delete operations run in background mode. There are two ways to determine if a purge or delete is currently active: If the response to a `ps -ef | grep nsr` command indicates that there is more than one `nsrindexd` process active, the purge operation has not yet completed.

When you purge or delete a volume, the client file indexes do not shrink automatically. Instead, the freed index space is used to allocate records that are added in the future. To reduce the size of the client file indexes immediately after you purge or delete index entries, run the following command:

```
# nsrck -C clientname
```

To reclaim the index space for all clients, change to the */nsr/index* directory and run the `nsrck -C` command.

Large indexes can take up to a few hours to compress with `nsrck`. For more details, refer to the following man pages: `nsrck`, `mminfo`, `scanner`, `nsr`, and `nsrmm`.

## ▼ How to Override a Retention Policy

Save sets are retained on volumes and in the media database until the save sets expire. Ordinarily, a save set expires and is recyclable when the save set and all save sets that depend on it for recovery pass their retention policies. However, you can explicitly specify an expiration date for a save set that overrides the retention policy. Dependency rules still apply, however, which means that a save set is not marked “recyclable” until all save sets that depend on it are also marked as recyclable.

To explicitly override the retention policy, enter the `save -e` manual backup command at the command line.

## Cluster Servers

Cluster servers allow Backup to migrate or employ failover between other nodes in the same cluster. Failover allows another node in the cluster to take over the operations from the first node. Failover continues with the last interrupted save set.

Backup can be used for backups in a DIGITAL TruCluster Available Server Environment (ASE). By installing Backup as a highly available application on each node in an ASE, Backup will have failover (relocate) capability with the cluster server.

In ASE, Backup recognizes a cluster environment during installation and allows you to install Backup as either a cluster server or as a non-cluster server. If Backup is to run as a cluster server, you must install Backup as a cluster server on each cluster member in the ASE. Refer to the *Solstice Backup 5.5 Installation Guide and Release Notes, DIGITAL UNIX Version* for complete details on how to install Backup in a cluster environment, as well as how to migrate an existing Backup server running DIGITAL UNIX to a cluster server.

### ▼ How to Configure a Cluster Server

After Backup is installed on DIGITAL TruCluster ASE servers, you must complete the installation by running the Installation Verification Procedure (IVP) and configuring the servers.

To complete the configuration, follow these steps:

1. **Log in to the cluster member that is running the Backup tape service.**

2. **Run the IVP to verify that the software is available on your Backup server:**

```
# setld -v subsetname
```

Modify the `/etc/hosts` file to add the name `nsrhost` as an alias to the official host name of the Backup server. This ensures that Backup will attempt to connect to the designated Backup server.

3. **From the Backup `nwadmin` GUI Server window, select Setup from the Server menu and add any cluster member that is not already listed in the Server resource's Administrator attribute.**

4. **Edit or create the `/nsr/res/servers` file to add the Backup tape service as well as each cluster member to the list of servers allowed to back it up.**

5. **Select Groups from the Customize menu to define a backup group.**

6. **Select Details from the View menu within the Groups resource and enable the Autorestart attribute.**

   a. **The Backup cluster server will take the identity of the Backup tape service regardless of which cluster member is currently running the Backup service.**

   b. **The first time Backup runs, it creates the client resource for the Backup tape service. Client resources need to be created manually for any cluster member to be backed up by the Backup tape service.**

   c. **If you are using an enabler code, enter it in the Registration window. If an authorization code is being used, you need to define the hostIDs in `/nsr/res/hostids`. Refer to the** *Solstice Backup 5.5 Installation Guide and Release Notes*, *DIGITAL UNIX Version* **and follow the steps in "Registering Backup Licenses for Cluster Server Failover" to define your hostIDs and to complete your registration.**

   d. **Add all cluster member hostnames to the server's Remote Access list.**

---

**Caution –** Only save sets that are part of a savegroup with the Autorestart attribute enabled will be restarted after the Backup tape is relocated.

---

   e. **Change the server's save set value from All to another value, such as the shared filesystem associated with the Backup tape service.**

## How Relocation of the Backup Service is Handled

Each time the Backup tape service is relocated, whether manually due to a system crash or automatically due to ASE placement policy enforcement, the ASE software will shut down all the Backup daemons on the cluster member running the Backup tape service. It will then restart `nsrexecd` on that system.

The ASE software will then relocate the Backup service to the assigned cluster member. The /nsr link will be redefined to point to the Backup tape service shared disk: /nsr -> /tape-service-name/nsr. For example, the link to the /nsr directory may be configured as follows:

- On the cluster member that is running the Backup service:

```
/nsr->/jupiter/nsr
/nsr.Backup.local->/var/nsr
```

- On each cluster member that is not running the Backup service:

```
/nsr->/nsr.Backup.local
/nsr.Backup.local->/var/nsr
```

ASE *will not* relocate the Backup service if it cannot access any of the Backup tape service devices or if the storage disk is busy (which could happen if a user has changed to a directory on that filesystem).

## Shutting Down and Restarting the Cluster Server

To shut down the Backup daemons on the cluster server, you must take the Backup tape service offline by running the ASE asemgr command. This stops all Backup activity.

To manually restart the Backup daemons on a cluster server, set the Backup tape service online by running the ASE asemgr command.

## ▼ How to Make a Cluster Member a Client of the Backup Cluster Server

When you install the Backup software on a cluster member, the installation creates the /nsr link to point to a local disk. It also creates a second link to the local Backup directory named nsr.networker.local. For example, if the local Backup directory was created in /var/nsr, after the installation, each client member will have the following links:

```
/nsr->/nsr.Backup.local
/nsr.Backup.local->/var/nsr
```

To make a cluster member a Backup client of the Backup cluster server, follow these steps:

1. **Using the** `nwadmin` **GUI, select Client Setup from the Clients menu to add the cluster member as a client of the Backup server.**

2. **Add the IP name for this cluster member's cluster interconnect to its Remote Access attribute list. For example, add** `root@mars.com` **to the Client resource for the client computer** `mars`**.**

3. **Add each cluster member to the server resource remote access list.**

# Backup Performance

Backup performance varies based on the specific network environment in which Backup operates, in addition to Backup settings. Several factors external to Backup enter into performance calculations, including CPU speed, speed of data storage devices, the limitations of the network, and the volume of network traffic.

Within Backup, you can tune performance by changing the following attributes:

- Parallelism

  This attribute in the Server resource sets the maximum number of savestreams that Backup allows to arrive at the server at one time. The maximum parallelism value is determined by the version of Backup you purchased.

- Target Sessions

  This attribute in the Devices resource sets the number of save streams that a storage device can manage and multiplex to a volume. Use this attribute to maximize the performance of each device.

You can increase backup speed by setting Backup to multiplex (or interleaf) data on a storage device. That is, data from more than one save set can be written to a single storage volume, and data from one save set can be spread across multiple volumes. (Each save set that is multiplexed must, by definition, belong to the same pool of storage volumes.) Multiplexing optimizes and distributes the flow of data from multiple clients to all the storage devices that are available to Backup. Upon recovery, however, performance can suffer because the data from one save set might have been written to several volumes.

Backup maintains the integrity of each save set’s data through a function that codes and tracks data by a *save set identification number* (SSID). The extent of multiplexing that can occur on any storage device is defined by the device’s value for Target Sessions.

It is often more efficient for Backup to multiplex multiple save sets to the same volume than to write each save set to a separate device. For this reason, Backup attempts to assign to each device a number of save sets up to the target value of sessions before assigning a save set to the next device.

# Event Notification

Backup reports activity and status to the system administrator through several different programs and interfaces. Backup uses *notifications* to determine which events to report and how to report on those events.

## Preconfigured Notifications

Backup includes many preconfigured notifications, which define the response of Backup to specific Backup events. The preconfigured notifications are described in "Notifications Resource" on page 24. You can edit the preconfigured notifications and create your own custom notifications. To view the events for which you can configure notifications, select Details from th View menu in the `nwadmin` GUI or Display Options from the Options menu, then select Hidden in the `nsradmin` interface.

### Routine Data Movement Operations Reports

The degree of success in the completion of scheduled group backups, group cloning, and archive operations is reported to you by the `savegrp` program through a *savegroup completion report*. (This report is the action invoked by the preconfigured notification called Savegroup Completion.) The report is e mailed to root and sent to the log file in `/nsr/logs/messages`. The report consolidates the following information:

- The success or failure of each save set participating in the operation
- The operation's save date and time
- The bootstrap SSID
- The bootstrap volume location (volume name, starting record number, and ending record number)

A second report, sent to the Backup server's designated default printer, repeats the bootstrap information as hard copy, which you should keep on hand in a secure location. (This printed report is the action invoked by the preconfigured notification called Bootstrap.) Disaster recovery is much easier to perform if you have access to the bootstrap information in the most recent printed report. Refer to the *Disaster Recovery Guide* for further information on using the bootstrap during disaster recovery.

The `nsrinfo` program enables you to query the contents of the Backup client file index. (Appendix B, “Command Line Reference Utilities,” on page 261 describes the most commonly used `nsrinfo` commands and options.)

The `nsrwatch` program enables you to use a character-based interface to monitor Backup activity as it occurs.

## Storage Management Application Reports

The following table lists the programs that Backup provides to query the contents of the storage management system. (Appendix B, “Command Line Reference Utilities,” on page 261 describes the most commonly used commands and options in more detail.)

**TABLE 3-5** Storage Management Report Programs

| Program Name | Function |
|---|---|
| `mminfo` | Generates a report that provides the contents and mode of the storage volumes, andthe identification numbers and status of the stored save sets. |
| `mmlocate` | Generates a report that provides the user-defined location of storage volumes. |
| `nsrinfo` | Generates a report that provides the contents of the client file index. |
| `nsrmm` | Generates a report that provides the status of the storage devices known to Backup. |

## Backup Server Statistics and Diagnostic Reports

Messages that report on Backup diagnostics are displayed in the Backup administrator interface and are also contained in the `/nsr/logs/messages` Backup messages file. These messages include warning and error conditions and notice of lost connections.

## Message Log Files

The messages generated by the Backup server daemons (nsrd, nsrindexd, nsrmmdbd, and nsrmmd) are contained in the Backup messages log and the daemon.log file, typically found in the /nsr/logs directory.

# Maintenance Tasks

This section contains tasks you might need to perform after you install and configure your Backup server.

## Message Log Management

Backup stores the messages generated by the Backup server daemons in a message log file in the /nsr/logs directory. When the log file becomes too large, delete some messages from the log. To automatically control the size of the log, you can use variables in the Backup startup script in the /etc directory, or create a script that uses the operating system services.

### ▼ How to Set the Startup Script to Trim Log Files

To modify the way that Backup services manage the Backup log files, change the following environmental variables in the Backup startup script, /etc/init.d/networker or /sbin/init.d/Backup, before you start the nsrd daemon:

- To change the maximum size of log files, change the NSR_MAXLOGSIZE value. The default value for NSR_MAXLOGSIZE is 1024 KB.
- To change the maximum number of log files that are saved, change the NSR_MAXLOGVERS value. The default value is 4.

Every time Backup starts, it checks the size of the daemon.log file. By default, when the daemon.log file reaches the 1024 KB limit, it is renamed daemon.001 and a new empty daemon.log is created. If the daemon.log file fills again, the names of each existing file shift so that the daemon.001 file is renamed daemon.002, daemon.log is renamed daemon.001, and a new empty daemon.log file is created. This process is repeated until the value in NSR_MAXLOGVERS is reached, at which point the highest numbered log is removed.

**Caution –** The trimming mechanism only functions when you start nsrd. The nsrd daemon does *not* check periodically to see whether the log file has exceeded NSR_MAXLOGSIZE. If nsrd runs for a long time, the log file can still grow very large. To activate the trimming mechanism, enter nsr_shutdown to stop the Backup daemons, then restart the nsrexecd and nsrd daemons.

## How to Use the Operating System Services to Trim Log Files

You can use the operating system services to automatically manage the size of the Backup log files. This example uses the operating system services available on the Solaris platform.

Solaris systems provide a two-part mechanism for managing the syslog message file (/var/log/syslog): a shell script (/usr/lib/newsyslog) and a crontab entry for root to periodically invoke the script.

You can modify the newsyslog script to manage and maintain a short history of the Backup server’s log file. The modified script maintains a three-file history of the Backup server‘s daemon.log file.

To manage your Backup log file, follow these steps:

1. **Use your favorite text editor to add the following lines to** /usr/lib/newsyslog*:*

```
LOGDIR=/nsr/logs
LOG=daemon.log
if test -d $LOGDIR
then
    cd $LOGDIR
    test -f $LOG.1 && mv $LOG.1 $LOG.2
    test -f $LOG.0 && mv $LOG.0 $LOG.1
    test -f $LOG   && mv $LOG   $LOG.0
    cp /dev/null $LOG
    chmod 644 $LOG
fi
```

Backup cannot use the new log file until you shut down and restart the Backup daemons. Shut down the daemons with the nsr_shutdown command, either manually or as an additional command in the newsyslog script. Make sure that the script does not run during a scheduled save.

For servers that run HP-UX, edit the /sbin/init.d/Backup file. Add the following line before the line that starts nsrd:

```
NSR_NO_PING=ok; export NSR_NO_PING
```

Then for Solaris, HP-UX, DYNIX/ptx, and AIX, restart Backup manually using the following commands:

- For Solaris and DYNIX/ptx:

```
/etc/init.d/networker start
```

- For HP-UX:

```
/sbin/init.d/networker start
```

- For AIX:

```
nsrexecd
nsrd
```

2. **Add an entry to** crontab **for root to control the frequency of running the** newsyslog **script. The entry shown in the following example invokes the** newsyslog **script every Saturday morning at 4:05 a.m., for example:**

```
5 4 * * 6   /usr/lib/newsyslog
```

If your Solaris system does not have the newsyslog script and crontab entry to invoke it, create the newsyslog script manually and add the crontab entry for it. See the crontab man page for details on creating crontab entries.

## ▼ How to Move Your Backup Server License to a Different Computer

1. **Perform a full backup of all the filesystems on the old Backup server.**

2. **Shut down the Backup daemons on the old server, using the** nsr_shutdown -a **command.**

3. **Make a tar tape of the entire /nsr directory from the old server, and reload it on the new server. If /nsr is a symbolic link on the old server, make sure the new server has the /nsr symbolic link set up also.**

4. **Shut down your old server and disconnect all the devices.**

5. **Shut down the new computer, add the hardware devices to the new server, and restart both computers. Start up the old computer first, and then the new one.**

6. **Install Backup on the new server.**

   If you have an autochanger, do not select the option to start the Backup daemons. Refer to the instructions in the *Solstice Backup 5.5 Installation Guide and Release Notes* to learn how to install and test the Backup device drivers.

   Because you created a new host, you must correctly define the index entry for the new host before you start the Backup daemons. There are two ways to correct the index entry:

   - Name the new server with the same host name as the old server at the operating system level before you modify client resources.
   - Create a new host name for the new server with the same configuration choices as the old server.

   To create a new hostname for the new server, follow these steps:

1. **Create a new host name for the new server with the same configuration choices as the old server.**

2. **Delete the host name entry for the old server.**

3. **Shut down the Backup daemons on the old server and the new server:**

```
# nsr_shutdown -a
```

4. **Change to the directory containing the old server index entry:**

```
# cd /nsr/index
```

   The entry for the new server hostname is empty.

5. **Delete the entry for the new server host name:**

```
# rmdir new-hostname
```

You must remove this entry, or the next step creates a subentry for the new server instead of the correct entry.

6. **Rename the old index directory to the new server host name:**

```
# mv old-hostname new-hostname
```

The Backup daemons start up on the new server.

You see the following messages on the new server:

```
new-server syslog: Backup Server: (notice) started
new-server syslog: Backup Registration: (notice) invalid auth
codes detected.
new-server syslog:
new-server syslog: The auth codes for the following licenses
enablers are now invalid.
new-server syslog: The cause may be that you moved the Backup
server to a new computer.
new-server syslog: You must re-register these enablers within 15
days to obtain new codes.
new-server syslog:
new-server syslog: License enabler #xxxxxx-xxxxxx-xxxxxx (Backup
Advanced/10)
```

Reregister your new Backup server. After you move Backup from one system to another, you have 15 days to register the new server with Sun. To reregister your new server, contact Sun Customer Service and request a Host Transfer Affidavit.You must then complete and return the Host Transfer Affidavit to Sun. You will then receive a new authorization code, which you must enter into the Auth Code field of the Registration window.

After you have successfully moved your server, follow these steps:

1. **Verify that all the clients are included in the scheduled backups.**

2. **Use the Backup** `recover` **program to make sure all the client indexes are visible and, therefore, recoverable.**

3. **Back up the indexes on the new server or perform a full backup of the new server as soon as possible.**

If you want to set up the old server as a client, first remove all the Backup software and the `/nsr` directory from the old server, then reinstall the Backup client software.

CHAPTER 4

# Device and Media Management

This chapter describes device and media operations you can perform through the Backup server. This chapter consists of the following sections:

## Device Configuration

A device is a drive that reads and writes data to storage volumes or disk files during backup, recovery, and other operations. The Devices resource contains the attributes for each device. The instructions for configuring your devices differ depending on whether the device is stand-alone or is contained in an autochanger or a silo.

For the Backup server to recognize your storage devices, you must configure each storage device individually.

If you use tape drives as your storage devices, you must use nonrewinding devices because Backup writes a file mark on the volume at the end of each backup and then appends data onto the volume based on the position of the file mark. If the device rewinds the media, the file mark position will be lost and previously written data is overwritten by the next backup. The pathnames for these devices must follow the Berkeley Storage Device (BSD) semantic rules, for example, `/dev/rmt/0mbn`. The "b" in the pathname satisfies the BSD semantics requirement.

If you use a file device, you must enter it as a directory path (the same as for other device types) rather than as just a file name.

# Storage Devices and Media Types Supported by Backup

Backup ships with the following list of supported storage devices and corresponding backup media types:

- 3480 tape drives (3480)
- 3570 tape drives (3570)
- 3590 high-speed tape drives (3590)
- 4890 tape drives (4890)
- 4-millimeter (DAT) tape drives (4mm)
- 4-millimeter 4 Gbyte tape drives (4mm 4GB)
- 4-millimeter 8 Gbyte tape drives (4mm 8GB)
- 8-millimeter tape drives (8mm)
- 8-millimeter five Gbyte tape drives (8mm 5GB)
- 9490 Timberline tape drives (9490)
- 9840 tape drives (9840)
- Digital linear tape drives (dlt)
- Digital linear 7000 tape drives (dlt7000)
- dlt tape drives (tz85, tz87, tz88, and tz89))
- dst high-speed tape drives (dst)
- dtf high-speed tape drives (dtf)
- Half-inch magnetic tape drives (himt)
- Quarter-inch cartridge tape drives (qic)
- SD3 high-speed tape drives (SD3)
- VHS tape drives (vhs)
- File system (file)
- Logical volumes (logical)
- Optical or magnetic disk drives (optical)

# Standalone Device Configuration

If you have a stand-alone device attached to the Backup server or storage node, display the Devices resource on the Backup server and enter or change the settings in that resource's attributes.

# How to Configure a File Device

To configure a disk file on the Backup server or storage node, follow these instructions:

1. **Display the Devices resource on the Backup server.**

2. **Enter the directory path of the file system you want to use in the Name attribute. This value must be a full, valid pathname, not a filename.**
3. **Select File for the Media Type attribute.**
4. **Leave the options for cleaning at their default settings so that automatic cleaning is not invoked.**
5. **Do not enable the Auto Media Management feature for file devices.**
6. **Save the configuration.**

   You label and mount the file device in the same manner as for other offline media used for Backup backups. Use a file as a device to facilitate save set staging (see “Save Set Staging” on page 93).

---

**Caution –** The name *path/tmpfs* is not allowed on Solaris servers.

---

## Autochanger Device Configuration

Machines such as autochangers and silos contain several devices. The way to configure devices in a machine that contains several devices involves a number of steps, which differ depending on whether the machine is an autochanger or silo.

To configure the devices in an autochanger, install and enable the Backup device drivers on the Backup server or storage node computer, then use the `jb_config` program to configure the autochanger and define the individual devices in the autochanger in the Devices resource. For detailed information about autochangers, see Chapter 7, “Autochangers,” on page page 153.

To configure devices in a silo for Backup to use, first install and enable the Backup Silo Support Module on the Backup server or storage node computer. Then use the `jb_config` program to configure the silo and its devices. Do not use the Devices resource to change or delete devices in a silo. See Chapter 11, “Silos,” on page page 237 for more details about silos.

## Hardware Compression Versus Software Compression

Backup client computers can compress data during backup, before the data is moved over the network or written to tape. You can implement software compression by selecting compression directives in the Clients resource or adding `compressasm` to a custom backup command. The `compressasm` feature typically achieves a 2:1

compression ratio. In addition to the performance advantages of moving less data across the network, software compression works better than some types of hardware compression in cases where a tape has a bad spot.

To handle end of tape (EOT) errors caused by bad spots on a tape, Backup maintains a fixed-size, write-behind buffer. When Backup requests the next tape, it flushes the write-behind buffer to the new tape. (EOT errors will not be handled if the size of the unflushed data is greater than the Backup buffer). The write-behind buffer has a finite size to handle noncompressing tape drives. This write-behind buffer also works with tape drives that compress data as it is written from the drive's buffer to tape, but not with drives that compress data as it is copied into the drive's buffer. The drive's buffer represents a ratio of 1.5 to 3 times as much data as it holds, byte for byte, and possibly much more (some drives claim compression ratios of 10:1). The write-behind buffer must be very large to handle a best-case 10:1 compression ratio possible with some drives. Real memory and swap space consumption make this prohibitive.

Use the following tips to decide which compression method is better for your environment:

- Use `compressasm` to minimize network bandwidth, if you have available CPU power.
- Use `compressasm` or compressing drives to get more data on a tape.
- Do not try to compress data that has already been compressed; you will obtain no additional compression. Although you can use compressing drives *and* `compressasm`, selecting both options might even expand the data.
- Do not use `compressasm` if you have a compressing drive and no networked clients.

# Storage Nodes, Remote Devices, and Multiple Network Interfaces

You can control most operations on local and remote devices, including autochangers and silos, from the Backup administration program on the server. But for some remote autochanger operations (for example, reset) you must use the `nsrjb` command or the `jb_config` program on the storage node computer. During data transfer operations, the Backup server uses remote devices the same way it uses local devices.

**Caution –** Backup clients for release 4.2 and later are able to use remote devices for backup, archive, and hierarchical storage management (HSM) functions. However, earlier Backup clients cannot back up data to remote devices.

This section also discusses network interfaces. You can change the default network interfaces. You can also direct different clients to different network interfaces into the same storage node.

## Remote Device Configuration

You configure remote stand-alone devices in an administration session with the controlling Backup server the same way you configure a stand-alone device that is connected to the Backup server. When you create each device, add a prefix to the device name that includes `rd=` and the storage node's host name. For example, `rd=omega:/dev/rmt/1mbn` creates a device called `/dev/rmt/1mbn` on a storage node computer called `omega`. For specific instructions, see the online help for configuring devices.

There are two steps to configure a remote autochanger or silo device. First, verify that the storage node is listed in the Administrator attribute in the Server resource of the controlling server. It must have the form *root@hostname*, where *hostname* is the host name of the storage node. Then run the `jb_config` program on the storage node computer to define each device in the autochanger or silo. See "`jb_config`" on page 308 or refer to the `jb_config` man page for the syntax and options for this program.

When the `jb_config` program is completed, you can remove the storage node's hostname from the Administrator list. If you add another autochanger later, you must add the storage node's host name to the Administrator attribute before running the `jb_config` program again.

## Multiple Network Interfaces

If you prefer to use an interface other than the default interface, use the server network interface attribute. Enter the preferred interface in the client's server network interface attribute.

When doing a `save`, you can have multiple network interfaces defined on a storage node. You specify the storage node's interfaces in the Storage Nodes attribute list. This allows different clients to use different network interfaces for the same storage node.

# Media Management

This section gives conceptual information about the media management features of Backup. You configure media management functions using the Backup GUI administration program (nwadmin), the nsradmin interface, or the nsrmm command. Detailed explanations of specific attributes are available in the online help. Refer to the nsradmin and nsrmm man pages for details concerning these Backup interfaces.

## Pools

A pool is a specific collection of media to which Backup writes data. Backup uses pools to sort and store data. The configuration settings for each pool act as filters that tell Backup which volumes should receive specific data. Backup uses pools in conjunction with label templates to keep track of which data is on which specific volume. For detailed information about label templates, see “Labeling Storage Volumes” on page 82.

### How Backup Uses Pools

The way you configure pools determines which volumes receive data. Each pool configuration contains a list of criteria that the data must meet for the data to be written to associated volumes. When you specify save sets to include in a pool, you can specify exact save set names, or you can use regular expression matching to send a group of save sets to a specific pool. For an example using regular expression matching, see “Example: Directing Client Indexes and Bootstrap to a Separate Pool” on page 78. For detailed information about regular expression matching, refer to the nsr_regexp man page.

When a scheduled backup occurs, Backup tries to match the save set to a pool configuration. If the save set matches the criteria of a pool configuration, Backup directs the save set to a labeled volume from that pool.

Backup then checks to see whether a correctly labeled volume is mounted on a storage device. If a correctly labeled volume is mounted on a storage device, Backup writes data to the volume. If a correctly labeled volume is not mounted on a storage device, Backup requests that such a volume be mounted and waits until an operator mounts the appropriate volume.

## Backup Pool Types

Backup provides preconfigured pool types to keep different types of data separate. Backup does not mix the following types of data on volumes within a pool:

- Backup data
- Archive data
- Cloned data
- Migration data

Unless you specify other pools, all backup data is routed to the Default pool and all archive data is routed to the Archive pool. Cloned backup data is routed to the Default Clone pool, and cloned archive data is routed to the Archive Clone pool.

## How Backup Uses Pool Criteria to Sort Data

When you configure Backup, you can create additional pools and sort data by pool type and any combination of the following criteria:

- Group (backup group)
- Backup client
- Save sets (file or filesystems)
- Backup levels (full, levels 1–9, incremental, manual)

If you begin by entering a group name in the Group attribute, the pool is immediately restricted to accept only data associated with the named group. If you add a second group name to the Group attribute, the pool accepts data associated with either group, but no others. Entries for a single attribute function as OR clauses; that is, the pool accepts data from clients in either group.

Each of the four configuration criteria, however, functions with the others as an AND clause. That is, if you enter configuration criteria in both the Group attribute and Save Set attribute, only data that meets both the Group criteria *and* the Save Set criteria is written to volumes from the specified pool.

You cannot create pools that share identical settings for pool type, group, client, save set, or level. If the settings for a new pool match the settings for an existing pool, you receive a warning message. Change the appropriate settings and reapply to save the pool resource.

For further information about save sets, see “Specifying Which Data Is Backed Up” on page 109. For further information about groups or backup levels, see “Backup Levels” on page 45.

## Example: Directing Client Indexes and Bootstrap to a Separate Pool

You can use regular expression matching to direct the client indexes and bootstrap to a different pool than you send the backup data.

In the following example, the client file indexes are in `/nsr/index`. To send the Backup server's bootstrap and all the client file indexes from this filesystem to the same pool, create a pool (in the Pools resource) with the following attributes:

```
name: Index;
pool type: Backup;
save sets: bootstrap, /nsr/index/.*;
levels: ;
```

When the group's scheduled backup runs, the client save sets are written to a volume labeled for the appropriate save set pools, while the Backup server's bootstrap and `/nsr/index` save sets are written to a separate volume labeled for the Index pool.

## When Data Meets the Criteria for More Than One Pool Configuration

Depending on the pool configurations you create, you might have data that matches the criteria for more than one pool configuration. For example, if you configure one pool to accept data from a group called Accounting and you configure another pool to accept data from all full backups, Backup has to determine to which pool a full backup for the Accounting group is written. Backup uses the following pool selection criteria:

1. Group (highest precedence)

2. Client

3. Save set

4. Level (lowest precedence)

When data matches the attributes for two pools, for example, Group and Level, the pool data is written to the pool specified in the Group attribute. For example, in the case where the data from the group matched the criteria for two different pools (the pool configured to accept data from the Accounting group and the other pool configured to accept data from all full backups) the data is routed to the pool that accepts data from the Accounting group.

## When Data Does Not Meet the Criteria for Any Pool

When you use customized pool configurations to sort your data, you might inadvertently omit a client or save set. During a scheduled backup, if data does not meet the criteria for any customized pool configuration, Backup automatically sends the data to the Default pool. Backup uses the Default pool to ensure that all data for clients in a backup group is backed up to a volume.

When Backup sends data to the Default pool, Backup looks for a labeled volume from the Default pool mounted on a storage device. If no Default pool volume is mounted on a storage device, Backup requests the appropriate volume and waits until an operator mounts the volume. If Backup asks for a Default pool volume in the middle of a scheduled backup but an operator is not present to mount a Default pool volume, the backup pauses until an operator mounts a Default pool volume. If you have an operator available to monitor the backups, it is a good idea to keep a volume labeled for the Default pool close at hand in case this situation arises unexpectedly.

If you plan to use Backup for unattended backups, run a test of the backup after making any configuration changes to ensure that all data is written to the appropriate volumes and to avoid an unexpected Backup request for a Default pool volume. For the procedure to test your scheduled backup, see “Immediate Start of a Scheduled Group Backup” on page 39.

## Configuring a Pool for Incremental Backups

If you want to create a separate pool for incremental backups, be aware that the Backup hierarchy of precedence affects the way the data is stored. If the Level attribute value is “incremental,” incremental data is routed to the associated pool but the corresponding changes to the client’s file index are not. Backup saves all client file indexes at level 9 to speed the recovery operation, if one is needed.

If the client file indexes *do not* meet the criteria for the pool associated with the incremental backups, Backup matches the indexes to *another* pool (usually the Default pool) and looks for an appropriately labeled volume to write to. If you need to recover your data, you might have to use a large number of volumes to recover all your data. Thus, to store the client file indexes along with the incremental backup data and to speed the recovery operation, define the Level value in the Pools resource to accept both level 9 and incremental data.

You can use the Backup preconfigured NonFull pool settings to ensure that the client file indexes belong to the same pool as their incremental backups. When you keep the indexes in the same pool as their incremental backups, you reduce the number of volumes you need for a recovery.

## Configuring a Pool for Clone Data

If you want to clone data, Backup requires a specific pool to receive the clone data, and a minimum of two devices, one to read the source volume and the other to write the clone. If you do not associate data to be cloned with a customized clone pool, Backup automatically uses the Default Clone pool. You must mount an appropriately labeled volume on a separate storage device for the cloning process to proceed smoothly. See "Cloning" on page 94 for more information on the Backup cloning feature.

## Configuring a Pool for Archive Data

If you want to use Backup Archive to archive data, Backup requires a specific pool to receive the archive data. You can then store these volumes offsite, if you want. If you do not associate data to be archived with a customized archive pool, Backup automatically uses the preconfigured Archive pool. You must mount an appropriately labeled volume on a storage device for the archive process to proceed smoothly. See Chapter 6, "Archive," on page page 237 for more information on the Backup archive feature.

## Configuring a Pool for Migration Data

If you use the HSM or XDSM HSM feature, Backup requires a specific pool to receive the premigrated and migrated save sets. If you do not associate the migration data with a customized migration pool, Backup automatically uses the preconfigured Migration pool. You must mount an appropriately labeled volume on a storage device for the premigration and migration processes to proceed smoothly. See Chapter 8, "Hierarchical Storage Management," on page 179 for more information on the Backup HSM feature.

---

**Caution –** Archive and migration data are both written in a different format than regular Backup save set data. Therefore, archive and migration data must each be written to different volumes. Because of these differences, the client file indexes and bootstrap save set created during a PC archive, premigration, or migration operation are also not written to the same volume as the archived or migrated save sets. By default, they are written to a volume from the Default pool. If you need to direct the client file indexes and bootstrap to a volume pool other than Default, see "Example: Directing Client Indexes and Bootstrap to a Separate Pool" on page 78 for information.

---

### Configuring a Pool for Manual Backups

You can create a customized pool to receive data from a manual backup by specifying “manual” in the Level attribute. Backup, however, sorts data from a manual backup differently than data from a regularly scheduled backup. Because a manual backup is not performed as part of a scheduled backup group, the data is not associated with any group name. Thus, when you perform a manual backup in which only a single client’s save set data is saved, the group normally associated with that client’s save set is not included as a criterion for pool assignment. As a consequence, data from a manual backup may be sent to a different pool than the pool in which data from this client’s save set is stored during a regularly scheduled backup operation.

If you do not create a customized pool to receive data from manual backups, Backup uses the Default pool and looks for a mounted volume from the Default pool on which to write manually backed-up data. Because Backup tracks the volume location of all backup data, you do not need to worry about tracking which volume contains the manually backed-up data. If you need to recover the data, Backup requests the correct volume.

---

**Caution –** When you perform a manual backup, the client index and server bootstrap are not included in the backup. If you never perform regularly scheduled backups of the clients and server computers, the information vital to data recovery in the event of a disaster is not available. Refer to the *Solstice Backup Disaster Recovery Guide* for further information on how the bootstrap is used during disaster recovery.

---

## Using Storage Devices and Pool Configuration to Sort Data

You can configure pools to sort data to different storage devices. You can either use specific media to receive data or designate a specific storage device to receive data from a designated pool.

### Volume Pools for Backup Data Directed to a Specific Device

You can associate a pool with a specific storage device. For example, you might want your full backups written to optical disk for off-site storage. You have two ways to ensure that data goes to one specific storage device:

- Keep a labeled volume associated with the appropriate pool mounted on a specific storage device. When a backup requires a volume with the correct label, Backup will find the volume on that storage device, which will be the only one available. If the volume resides in an autochanger, Backup automatically mounts the volume when it is requested.
- In the Pools resource, associate the pool with the device in the pool configuration attribute list. All data is written only to that device.

### Volume Pools for Backup Data Written to Different Media Types

You can write data across several volumes of different media types (for example, magnetic disk and tapes), as long as the volumes mounted on the storage devices have the appropriate label associated with the pool.

## Labeling Storage Volumes

Backup labels (initializes) each storage volume with a unique internal label that corresponds to a pool. During backup and other operations, Backup can identify the pool to which a volume belongs by its label. Backup applies a label template to create a unique internal label for each volume.

Backup uses label templates and pool configuration settings to sort, store, and track data on media volumes. If you need to recover data, Backup prompts you for the specific volume that contains the required data, by volume name and sequence number.

### How Backup Uses Label Templates

Backup writes a given set of data to a specific pool. For Backup to recognize that a particular volume belongs to the correct pool, the volume must have an internal identification label that associates it with the correct pool. The contents of the volume label follow rules defined in a specific label template that you create in the Label Templates resource. You then associate a label template with a specific pool in the Pools resource. If you do not associate data with a specific pool, Backup uses the preconfigured Default pool and corresponding Default label template. FIGURE 4-1 on page 83 illustrates how a pool configuration uses its associated label template to label a volume. You must configure a label template before you configure the associated pool for your custom template to be available in the Pools resource.

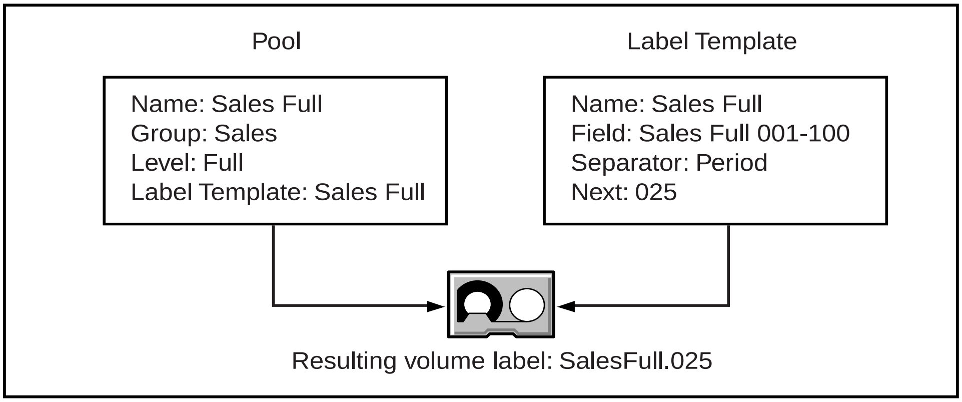


**FIGURE 4-1** How Backup Labels a Volume Using a Label Template

## Customizing Label Templates

To customize label templates, display the Label Template resource and specify values for the following attributes:

- Name

  Keep the label Name consistent with the pool Name so you and your users can easily see how the data is organized. You can use the same or similar names. For example, you can create a label template called "AcctFull" for volumes that belong to a pool called "Accounting Full."

  You can only use alphanumeric characters when you create label templates. Backup does not allow the following characters in label template names:

  / \ * [ ] ( ) $ ! ^ ' ; ' ~ < > & | { }

  In addition, you cannot use the following four characters, because they are used as separators in label templates:

  - Colon (:)
  - Dash (-)
  - Period (.)
  - Underscore (_)

- Fields

  A label template is made up of one or more fields. Each field, or component, provides a layer of specificity to your organizational structure. You can have as many components as you want, but it is best to keep the template simple, with few components. The total length of the label cannot exceed 64 characters.

  You can use four types of components:

  - Range of numbers (for example, *001-999)*
  - Range of lowercase letters (for example, *aa-zz)*
  - Range of uppercase letters (for example, *AA-ZZ)*
  - Character string (for example, *Accounting)*

Each range includes a start value, a dash (-), and an end value. The start value and the end value must have the same number of characters. For example, use “01-99,” not “1-99,” or “aaa-zzz,” not “aa-zzz.” (This rule does not apply to a list of character strings or words; character strings are separated by a blank space.)

The order in which you enter each component of the Fields template is important. Backup applies each component in a left-to-right order, starting with the first one you enter. Table 4-1 illustrates how label templates use components to create a number sequence for volume labels.

**TABLE 4-1** Examples of Number Sequences for Volume Labels

| Type of Components | Fields | Number Sequence Result | Total Number of Labels |
|---|---|---|---|
| Range of numbers | 001-100 | 001, 002, 003, ... 100 | 100 |
| Character string<br>Range of numbers | SalesFull<br>001-100 | SalesFull.001, ... SalesFull.100 | 100 |
| Range of lowercase letters<br>Range of numbers | aa-zz<br>00-99 | aa.00, ... aa.99,<br>ab.00, ... ab.99,<br>ac.00, ... ac.99,<br>:<br>az.00, ... az.99,<br>ba.00, ... ba.99<br>:<br>zz.00, ... zz.99 | 67,600 ($26^2$ times $10^2$) |

Your label template should allow for expansion of your backup media storage system. For example, it is better to create a template for 100 tapes and not use all of them than to create a template for only 10 tapes and run out of labels. When Backup reaches the end of the template number sequence, Backup wraps around to the start value. In Table 4-1, for example, after Backup uses zz.99 for the 67,600th label, Backup uses aa.00 for the 67,601st label.

- Separator

  Choose which character symbol you want to appear between component entries. You can use the period, dash, colon, or underscore to separate each component of the label template. If you do not select a separator, the label components do not have separators (for example, AA00aa), which makes the labels difficult to read.

- Next

  Choose the next sequence number to write on the label Backup places on a volume (according to the template). If you want to force a label to start the label scheme at a particular point, enter the start label value you want. Backup continues to generate labels from that point on, according to the rules of the template. If you want Backup to generate the first label for you, leave this attribute blank.

When Backup recycles a storage volume, the volume label does not change as long as the volume remains in the same pool. That is, if a storage volume labeled “Dev.006” is recycled, it retains the volume label “Dev.006” and does not receive a new label with the next sequence number.

## Using Label Template Components

Backup is shipped with preconfigured label templates that correspond to the preconfigured pools. If you choose to create your own templates, you can include as many components in the Fields attribute as necessary to suit your organizational structure. However, it is a good idea to keep the template simple with few components. For example, if you create a label template for your Accounting Department, you can customize your label template in several ways, depending on the size of your storage system and media device capabilities. The following table illustrates several ways you can use components to organize your labels.

**TABLE 4-2** Using Label Template Components

| Type of Organizational Structure | Fields (Components) | Separator | Resulting Volume Labels |
|---|---|---|---|
| Sequential | AcctFull<br>001-100 | Period | AcctFull.001<br>(100 total labels) |
| Storage oriented (for example, 3 storage racks with 5 shelves each, each shelf holding 100 tapes) | 1-3<br>1-5<br>001-100 | Dash | 1-1-001<br>This label is for the first tape in rack 1 on shelf 1.<br>(1,500 total labels) |
| Two-sided media (for example, optical devices) | AcctFull<br>000-999<br>a-b | Underscore | AcctFull_000_a (side 1)<br>AcctFull_000_b (side 2)<br>(2,000 total labels) |

# Storage Management Operations (Labeling and Mounting)

The internal label on a volume contains a unique name that Backup uses to track and recognize storage media. In the media database, Backup refers to volumes by their volume labels. Backup uses the media database records to determine which volumes are needed for backing up or recovering data.

Every volume belongs to a pool. Each pool has a matching label template associated with it. Volumes are labeled according to the rules of these label templates. Label templates provide a way to consistently name and label volumes so you do not have to track the number of volumes you have used. You can use the preconfigured pools and preconfigured (and associated) label templates that come with the Backup product, or create your own pools, label templates, and pool/template associations. Customizing your own label templates gives you more control over your data storage organization.

When you put a new internal label on a volume or relabel a volume to recycle, any existing data stored on the volume under the previous label is no longer available for recovery.

## Backup Criteria for Volume Selection and Mounting

When a scheduled or manual backup occurs, Backup searches for a volume from the appropriate pool to accept the data that needs to be written. The storage volumes available for Backup to use are the volumes that are mounted on stand-alone devices and the volumes accessible to Backup through auto media management or available to Backup through an autochanger or silo.

If you try to back up files when an appropriate volume is not mounted, Backup requests a writable volume by displaying a message similar to the following in the Pending display:

```
media waiting: backup to pool ‘Default’ waiting for 1 writable
backup tape or disk
```

When you start a data recovery, Backup displays a message in the Pending display that requests a mount of the volume name that contains the backed-up data:

```
media waiting: recover waiting for 8mm 5GB volume-name
```

If you need more than one volume to recover the files, the Pending display lists all of the volumes in the order they are needed. During the recovery process, Backup requests the volumes it needs, one at a time.

If you mount more than one volume on the storage devices used by Backup, Backup uses the following hierarchy to select a volume on which to write data:

1. An already mounted, appendable volume from the appropriate pool
2. An already mounted, recyclable volume from the appropriate pool that is not currently in use
3. An already mounted, unlabeled volume that is not currently in use and is in a device for which auto media management is enabled
4. An appendable volume that is not currently mounted in the device but is from the appropriate pool
5. A recyclable volume that is not currently mounted in the device but is from the appropriate pool

## ▼ How to Label a Volume

A volume label is a unique internal code applied by Backup that initializes the volume for Backup to use and identifies a storage volume as part of a specific pool. To label a volume, follow these steps:

1. **Place an unlabeled or recyclable volume in the Backup storage device.**
2. **Use Backup to label the volume. You can use either the Backup administration program or the `nsrmm` command. There are three options**:
   - If you do not select a pool for the volume that you are about to label, Backup automatically applies the label template associated with the Default pool.
   - To create individual label names not associated with a template, edit the Volume Name attribute in the Label resource and enter a unique label name.
   - If you enable the Manual Recycle attribute when you label a volume, the volume cannot automatically be marked as recyclable according to the retention policy. Only an administrator can mark the volume recyclable.

When Backup labels a volume, Backup first verifies that the volume is unlabeled. Then Backup labels the volume with the name specified in the Volume Name attribute, using either the next sequential label from the label template associated with the chosen pool or an override volume name you entered.

If you relabel a recyclable volume from the same pool, the volume label name and sequence number remain the same, but access to the original data on the volume is destroyed and the volume becomes available for new data.

After a volume is labeled and mounted in a device, the volume is available to receive data. Because the Backup label is internal and machine-readable, it is a good idea to put an adhesive label on each volume that matches the internal volume label. To use barcode labels with an autochanger, see "How Backup Uses Barcode Labels With Autochangers" on page 170. To use barcode labels with a silo, see "How to Label Volumes in a Silo" on page 245.

## ▼ How to Mount or Unmount a Volume

When you issue the command to mount a volume or when Backup mounts a volume through auto media management, a volume that is loaded in the storage device is prepared to receive data from Backup. For example, when a tape is mounted, the read/write head of the device is placed at the beginning of the blank part of the tape, ready to write.

To mount the volume in the device, you can use either the Backup administration program or the command line:

- In the Backup administration program, select the device from the Devices display, then click the Mount button.
- At the shell prompt, enter the `nsrmm` command with the `-m` option.

After you label and mount a volume, the volume name is displayed in the Devices resource beside the *pathname* of the device in the Backup administration program.

To perform an unattended backup using a stand-alone device, you must mount labeled volumes in the device before leaving it unattended.

---

**Caution –** You can only use nonrewinding devices with Backup. If you use a rewinding device, the read/write head is repositioned at the beginning of the volume and the previously backed-up data is overwritten.

---

# Timeout Settings for Remote Devices

You can time out a mount request on a remote device storage node and redirect the save to another storage node. Set the attributes Save Mount Timeout and Save Lockout in the Devices Resource to change the timeout of a save mount request on a remote device. If the mount request is not satisfied by the number of minutes specified by the Save Mount Timeout attribute, the storage node is locked out from receiving saved data for the number of minutes specified by the value of the Save Lockout attribute. The default value for Save Mount Timeout is 30 minutes. The default value for Save Lockout is zero, which means the device in the storage node continues to receive mount requests for the saved data.

---

**Caution –** The Save Mount Timeout only applies to the initial volume of a save request.

---

# ▼ How to Find a Volume Name

If the adhesive label on the volume is missing or illegible, you can determine the volume's name from the Backup administration program or the command line:

- In the Backup administration program, complete one of the following:
  - Mount the volume and view the volume name in the Devices display.
  - Start a labeling operation and view the Volume Name field in the Label resource, then cancel the operation.
- At the shell prompt, enter the `nsrmm` command with the `-p` option to show the label of the volume loaded in a device.

# How Backup Selects a Storage Volume for Relabeling

Backup data is destined for volumes from a specific pool. When the data is ready to be written, Backup monitors the active devices to locate a volume from the appropriate pool.

If only one volume from the pool is mounted and appendable, the data is directed to that volume.

If two volumes from the same pool are mounted on devices, Backup considers the following factors to guide its volume selection:

- The expiration date of the volume

By default, the volume expiration date is set at two years after the date on which the storage volume was labeled (or relabeled). To override this default setting, change the volume's expiration date in the Devices resource. If the default setting is overridden, Backup checks to ensure that the volume expiration date is farther in the future than the date on which the save set is set to exceed its retention policy. If Backup finds that the volume expiration date is later than the retention policy, then the save set is written to the volume. Otherwise, Backup does not write the save set to the volume. (This checking does not occur if the expiration date of the volume is not overridden.)

- The volume mode
  - If a mounted, appendable volume from the appropriate pool is available, Backup writes to it.
  - If there is no appendable volume of the appropriate pool available (and if you enabled auto media management), Backup recycles and then writes to a mounted, recyclable volume from the appropriate pool as a second choice. (Backup does not consider writing to a mounted, recyclable volume that belongs to a different pool.)
  - If no volumes of the pool are available (and if you did not enable auto media management), Backup labels, mounts, and writes to a new, unlabeled volume or a volume that does not have a Backup label.
- The volume label time, which is the time when the volume was labeled

  Volumes with the oldest label time are selected before volumes that were labeled more recently.
- The number of sessions currently writing to the device

  If Backup cannot find a mounted volume from the appropriate pool, a mount request is initiated. If auto media management is not enabled or if Backup has only stand-alone devices available, mount requests continue to be generated until a volume is mounted and backup can begin.

## Auto Media Management

The auto media management feature gives Backup automatic control over media loaded in the storage device. If you enable the auto media management feature in the Devices resource, Backup automatically labels, mounts, and overwrites a volume it considers unlabeled, and automatically recycles volumes eligible for reuse that are loaded into the device. The auto media management feature is only enabled for stand-alone devices in the Devices resource. To enable auto media management for devices in an autochanger, see "Auto Media Management With Autochanger Devices" on page 164.

Backup considers a volume unlabeled in the following conditions:

- The volume has no internal label.
- The volume is labeled with information other than a recognizable Backup label.
- The volume is labeled with a Backup label, but the density indicated on the internal label differs from the density of the device where the volume is mounted.

Because the auto media management feature can relabel a volume with a different density, it is possible to inadvertently overwrite data that still has value. For this reason, be careful if Backup volumes are shared between devices with different densities.

If you do not enable the auto media management feature, Backup ignores unlabeled media and does not consider it for backup.

If you enable the auto media management feature for a stand-alone device, the following processes occur when a volume becomes full during a backup:

1. A notification is sent that indicates that the server or storage node is waiting for a writable volume. At the same time, Backup waits for the full, verified volume to be unmounted.

2. The device is monitored and the software waits for another volume to be inserted into the device.

3. After a volume is detected, a check is performed to determine whether the volume is labeled. If it is already labeled, the volume is mounted into the device. Backup checks to see whether the newly-mounted volume is a candidate to write data to. If so, the write operation continues. If not, Backup continues to wait for a writable volume to continue the backup.

4. If the volume is recyclable and is a member of the required pool, it is recycled the next time a writable volume is needed.

5. If the volume is unlabeled, it is labeled when the *next* writable volume is needed for a save.

In general if a *non-full* volume is unmounted from a stand-alone drive and you enabled auto media management, Backup waits 60 minutes before it automatically remounts the volume in the drive. This hour is considered a reasonable delay to give you or an operator time to unload the volume after unmounting.

---

**Caution –** Backup considers volumes that were labeled by a different application to be valid relabel candidates if auto media management is enabled. Once Backup relabels the volume, the previously stored data is lost.

---

# Storage Volume Status

Different reports and windows provide information on the status of storage volumes using parameters such as Written, %Used, Location, and Mode. This section defines some of the most common terms contained in reports about volumes.

In the Backup administration program, the volume name displayed is the same as the name that appears on the volume label. At the end of the volume name, the following designations might be displayed:

- A, which indicates an archive storage volume
- R, which indicates a volume that is considered read-only

The value of Written always indicates the exact number of bytes written to the volume.

The value of %Used is based on an estimate of the total capacity of the volume, which is derived from the specified value of the Media Type of the device resource. Backup does not use the value of %Used to determine whether to write to a volume. Even if a volume is marked 100% used (a %Used value of 100% means that the value of Written is equal to or exceeds the estimate for the volume), Backup continues to write to the volume until it is marked full. Backup marks a volume full when it reaches the end of the media or encounters a write error.

The storage volume location refers to a character field you define in the Volumes resource that describes a physical location meaningful in your environment, such as “2nd shelf, Cabinet 2, Room 42.“

Table 4-3 lists all the possible storage volume modes and their definitions within Backup.

**TABLE 4-3** Storage Volume Modes

| Mode Value | Meaning | Description |
|---|---|---|
| `appen` | Appendable | This volume contains empty space. Data that meets the acceptance criteria for the pool to which this volume belongs can be appended. |
| `man` | Manual recycle | This volume is exempt from automatic recycling. The mode can only be changed manually. |
| `(R)` | Read-only | The save sets on this volume are considered read-only. The mode can only be changed manually. |
| `recyc` | Recyclable | This volume is eligible for automatic recycling. Before the volume can be overwritten, it must first be relabeled. |

In general, a storage volume becomes recyclable when all the individual save sets located on the volume have assumed the status of Recyclable. For more information about save set status, see “Save Set Status Values” on page 119.

## Save Set Staging

Save set staging is a process of moving data from one storage medium to another and removing the data from its original location. If the data was on a file device type, the space is reclaimed so that the disk space can be used for other purposes. You use save set staging to move save sets that you have backed up, archived, or migrated. Staging is especially recommended for save sets that you backed up to a file device type to move the data to more permanent storage, such as an optical or tape volume.

You can configure policies in the Stage resource to have Backup perform automatic staging once criteria you set is reached. Or you can use the `nsrstage` program to perform staging manually.

When you issue the `nsrstage` command, Backup creates a clone of the save set you specify on a clone volume of the medium you specify. If you stored the save set on a file device type, Backup deletes the save set from its original location to free the space the save set occupied. Backup tracks the location of the save set in the media database. The retention policy for the save set does not change when the data is staged.

To stage a save set using the command line, enter the `nsrstage` command at the shell prompt. For example, to stage an individual save set, enter the following command:

```
# nsrstage -s server -b pool -m -S save-set-ID
```

Refer to the `nsrstage` man page for the syntax and options for the `nsrstage` program.

To set or change staging policies, choose one of the following methods:

- Use the `nsradmin` command to start the cursor version of the administration program (select the NSR Stage resource), *or*
- Use the `nwadmin` command to start the GUI version of the administration program (select Customize>Staging).

One staging resource, named “default stage,” is shipped with the software. This resource has the attribute values shown in the following table. You can make changes to the default stage resource, but you cannot delete it.

Refer to the online help available in both the `nsradmin` and `nwadmin` programs for more details about the attributes available in the Stage resource.

**TABLE 4-4** Stage Resource Preconfigured Attribute Settings

| Attribute | Preconfigured Settings |
|---|---|
| Enabled | No |
| Max Storage Period | 7 days |
| High Water Mark (%) | 90 |
| Low Water Mark (%) | 60 |
| Save Set Selection | Oldest Save Set |
| Destination Pool | Default Clone |
| Devices | The device you specified during installation |
| Recover Space Interval | 8 |
| File System Check Interval | 3 |
| Start Now | Null |

The Start Now attribute enables you to start one of three operations immediately after you save changes to the resource, regardless of whether the conditions you set for the staging policies are present:

- Recover space
- Check the filesystem
- Stage all save sets

Once a save set is staged from a file device, its media database entry for the original file volume is removed, and the space once occupied on the file volumes can be freed and recovered for new save sets.

# Cloning

Cloning is a process of reproducing complete save sets from a storage volume to a clone volume. You can clone save set data from backups, archives, or migration. You can clone save sets automatically (as part of a backup, archive, or migration operation) or manually at another time.

Use cloning for higher reliability or convenience. For example, you can store clones offsite, send your data to another location, or verify backed-up data.

The cloning operation happens in two steps: First, Backup recovers data from the source volume. Then, Backup writes the data to a clone volume (a volume from a pool of type “clone”). Cloning requires at least two active devices, because one is required for reading the source volume and one is required for writing the new, cloned data. During cloning, the reproduction of data is from source volume to clone volume. Cloning does not involve data stored on the clients or server. Backup allows only one clone of a save set per volume. Therefore, if you specify three clones of a save set, each is written to a separate volume.

Automatic cloning (that is, cloning associated with a scheduled group backup operation) is performed after all backup operations are complete. The savegroup completion report that is issued after a scheduled backup also includes a report of the success or failure of the cloning operation for each save set.

The location of the devices where the clone data is written is determined by the list in the Storage Nodes attribute in the Clients resource for the Backup server. You can add or remove the names of storage nodes and the Backup server at any time, but you cannot have a different list of storage nodes to receive clone data than to receive backup data.

If you want to perform cloning long after a group has finished, you must do the cloning manually, volume by volume, or from the command line using a script in combination with a batch file. If you execute cloning manually, no report is generated.

When you clone data, different capacities of storage media may mean that more or fewer clone volumes are required. The cloning operation leaves traceable information entries in both the client file index and the media database. The capability to track cloned data distinguishes cloning from an operating system or hardware device copy operation.

To initiate cloning for a complete scheduled backup operation, enable cloning as part of the Group configuration. To clone individual save sets or clone single storage volumes, use the Save Set Clone or Volume Clone windows in the `nwadmin` GUI, or the `nsrclone` program from the command line.

When you specify that a particular volume be cloned, Backup uses the save sets on the specified volume as the source data.

When you specify a clone of a particular save set, Backup determines whether the save set already has a clone. If multiple clones of a save set exist, clones of save sets on volumes in an autochanger are generally selected as the source data, rather than a volume that requires human intervention to mount. Command line options enable you to specify the precise save set clone to use as the source, if you want.

If you execute a clone operation manually, no completion report is generated. Messages generated by the `nsrclone` program are displayed in a message window in the administration program’s GUI and are also logged to the `/nsr/logs/messages` Backup message file.

## Clone Storage Node Affinity

The link between a client's resource of a storage node and a list of available storage nodes to receive cloned save sets from the storage node client is called *clone storage node affinity*. Data is cloned from media that contains the original save sets to media on the specified clone storage node. You define clone storage node affinity in the Clone Storage Nodes attribute, which is found in the Clients resource of a storage node. When you make a change to the Clone Storage Nodes attribute in the Client resource for a storage node client, the changed value is propagated to any additional Clients resources configured for that storage node client.

The Clone Storage Nodes attribute allows you to specify a different network interface for storage nodes that perform cloning operations than the network interface you specify for the storage node's remote device.

The server utilizes the exact host name you specify in the Clone Storage Nodes attribute, instead of using the host name prefix for the remote device name configured in the Devices resource.

When a volume is being cloned, the Backup server checks the value of the Clone Storage Nodes attribute for that storage node client. If the Clone Storage Nodes attribute has a null value, then the value listed in the server's Clone Storage Nodes attribute is used. If that list also contains a null value, then the server's Storage Node attribute is used. Compatibility is maintained with the existing clone function which follows the server's Storage Node attribute.

To independently direct clones from each storage node, add the host name of the storage node that you want to receive the directed clones to the Clone Storage Nodes attribute in the Client resource configured for the storage node. The first entry made on the list that has a functional, enabled device is selected to receive the cloned data from the storage node.

To direct clones from all storage nodes to the same destination, leave the Clone Storage Nodes attribute blank for the Clients resources you configure for the storage nodes, and only configure the Backup server's Clone Storage Nodes attribute. This tactic provides a single source of control for clone destination.

The file index and media database entries for the save sets cloned to media on a remote device on a storage node still reside on the Backup server, which enforces the browse and retention policies in the same manner as for any cloned save sets that reside on the media in a device that is locally attached to the server.

## Cloning Versus Duplication of Volumes

When you clone a volume, the volume is not simply duplicated. Each save set on the volume is reproduced completely, which could mean that more or less space is used on the clone volume than on the source volume.

You might prefer to make exact copies (duplicates) of Backup volumes to provide additional disaster recovery protection. This approach, which in UNIX relies on the `tcopy` command, is not recommended but might serve a specific environment adequately. If you rely on an exact copy command, you must first ensure that the destination volume can hold the number of bytes that are contained in the source Backup volume. In addition, be aware that Backup would have no knowledge of the duplicated volume since the volume is not entered into the server's media database. If you enabled automated media management and you leave the volume in an autochanger managed by Backup, the volume may be considered eligible for relabeling and use during a scheduled backup, because it does not have a valid Backup label.

Similarly, it is possible to make an exact copy of an archive volume. However, the annotation that is associated with each archive save set is information that is stored in the Backup server's media database, not on the volume itself. Therefore, a duplicate volume of the archived save set does not include the annotation. If the entry of the original archive save set is removed from the media database, the annotation that describes it is also removed.

## Cloning and Data Tracking Information

The clone operation does not insert entries into the client file index. Cloned save sets are only tracked through the media database. During a clone operation, the location of a cloned save set is added to the existing save set entry in the media database. That is, each save set clone shares the same SSID as the source save set. All characteristics that are true for the source save set are also true for the clone save set. If the source save sets are still browsable, the clone status is also browsable. If the source save sets have passed their browse policies, the clone status is recoverable.

Volumes that belong to a clone pool are also tracked through volume entries in the media database. The fact that all save sets share the same media database save set entry has implications for the following actions, which are executed on a "per save set basis" and not on a "per volume" basis:

- Changing the mode of a cloned volume (of save sets)
- Purging a volume (of save sets) from the client file index
- Deleting a volume (of save set locations) from the media database

**Caution –** If you manually change the mode of a cloned volume to “recyc” with the intent of reusing a particular clone volume, be aware that the mode of a volume only changes to recyclable when all the save sets on that volume are recyclable. Therefore, when the mode of the volume changes to “recyc,” you *effectively change the status of all save sets* on the clone volume to “recyc.” Because the save sets share the same entry in the media database, there is no distinction between original and clone save sets. The end result is that *all the save sets that reside on the now-recyclable volume or on any other volume* become candidates for immediate recycling.

To prevent inadvertent loss of data, if you want to reuse a particular clone volume and still protect the instances of a save set that exist on other volumes, first change the mode of the volumes to be protected to “man_recyc.” This means that Backup cannot automatically recycle the volume. Then, you can safely change the volume that you intend for reuse to mode “recyc.”

Similarly, if you *purge* a clone volume, you effectively remove from the client file index all file entries associated with all save sets that reside (in whole or in part) on the particular clone volume.

If you *delete* a clone volume, the `nsrim` index management program locates the entry in the media database for each save set that resides on the clone volume. From the entry, the `nsrim` program marks for deletion the information about the location of one of the save set clones from the entry. This action is performed for each save set entry. In addition, `nsrim` marks the entry for the particular clone volume (identified by its volume ID number) for deletion from the database.

## Cloning Performance

In general, a volume write that occurs as part of a backup operation and a volume write that occurs as part of a cloning operation proceed at the same speed. However, if a clone operation is automatically requested as part of a scheduled backup, you might experience a performance degradation in other scheduled backups that follow. Backup generally attempts to complete one group’s scheduled backup before a scheduled backup is initiated for another group. However, Backup considers that a group backup is finished when the *backup* operations are complete, not when any automatic cloning is complete. Therefore, if another group starts its backup while the previous group’s clone operation is underway, you may experience contention for `nsrmmd` resources or specific volumes. To avoid this problem, you may decide to refrain from automatic cloning and instead initiate a single clone operation by passing a set of SSIDs to `nsrclone` as part of a job that runs at a nonpeak time after all backups are complete.

# Cloning and Recovery

A clone volume is used for recovery any time Backup attempts to recover a particular save set and either the original save set volume has been deleted or the status of the original save set is marked “suspect.”

You can always execute the `scanner` program on a clone volume to rebuild entries in the client file index, the media database, or both. After you re-create the entries, traditional recovery is available. Refer to the *Solstice Backup Disaster Recovery Guide* for information on how to recover data with the `scanner` program.

CHAPTER 5

# Client Operations

This chapter describes how to configure and use Backup clients, and gives suggestions on how to best customize your client configurations to suit the needs of your environment. The following topics are addressed in this chapter:

- What Is a Backup Client?
- Configuring a New Backup Client
- How Backup Enforces Client Licensing
- Backup and Recovery With Backup Client Applications
- Archive and Retrieve Backup Client Applications

## What Is a Backup Client?

Backup clients are computers that connect to the Backup server for backup, recovery, and other operations. You can install Backup client software on the client computer or access it across the network. Backup client software is available for a variety of platforms; you can back up data from clients on a variety of platforms to Backup servers on a variety of platforms.

The basic Backup client software contains backup and recovery capability for filesystem data. You can purchase additional modules to enable archiving, HSM, and backup of a variety of databases.

### Cluster Clients

A cluster client shares resources mapped to another node in the cluster. Resources are shared among the cluster client members or nodes. A cluster client does not have the failover capabilities of a cluster server. See “Cluster Servers” on page 60 and refer to the *Solstice Backup 5.5 Installation Guide and Release Notes*, for more information.

# Configuring a New Backup Client

After you install the client software on the Backup client computer, you create a client resource on the Backup server that contains your configuration choices for each Backup client. These choices determine what data is backed up, according to which schedule, and whether additional features, such as archiving, are enabled.

## ▼ How to Create a New Client

To create a new Backup client, display the Clients resource in either the Backup administration program (`nwadmin`) or the `nsradmin` interface. Select the Create option and enter the hostname of the client computer.

If you choose not to customize the configuration choices, the new Backup client is automatically assigned the default configuration, after you apply and save the changes. (See "Clients Resource" on page 16 for more information about the default configuration for Backup clients.) The default setting (All) for the Save Set attribute means that all the files on the client are backed up during a scheduled or manual backup. (See "What Is a Save Set?" on page 109 for information about client save sets.)

Refer to the Backup online help for specific information about each of the attributes you can configure in the Clients resource. Refer to the `nsradmin` man page for more information on how to use the `nsradmin` interface to create, edit, and delete Backup resources.

# Backup Clients of Different Platforms

The Backup server can back up clients from a variety of platforms. This section provides configuration tips for configuring clients to enable them to back up to the Backup server.

To use clients of an operating system than is different from that of your Backup server, you must purchase and enable the appropriate ClientPak. See "How Backup Enforces Client Licensing" on page 132 for information about ClientPaks and how the Backup server checks each client before it allows backup to begin.

Support for 64-bit filesystems exists for clients that run Solaris 2.6, AIX 4.2, and HP-UX 10.20. You can archive, back up, browse, and recover files larger than two gigabytes for clients of Solaris 2.6, AIX 4.2, and HP-UX 10.20. If your clients are not 64-bit capable, you can browse files larger than 2 gigabytes, but you cannot recover them.

## UNIX Clients

On all Backup clients for UNIX, you must manually update and verify certain files and paths, as follows:

- The /etc/hosts file must contain the Internet address of the Backup client and the Backup server, unless you use DNS or Network Information System (NIS), for example:

```
127.0.0.1   localhostloopback
137.69.8.1  serverserver.companyname.com
137.69.8.2  client  client.companyname.com
```

  Backup does not automatically configure and update the /etc/hosts file. You must manually edit the file and verify that the information in this file is accurate. Do not delete or comment out the entry for the localhost loopback.

- During installation of the SunOS, Solaris, AIX, and DYNIX/ptx client software, if you accepted the default directory when installing the Backup executables, the default directory should already be in your executable path. If you specify a different directory, add the directory to your executable path for root or Backup users.

When you install the HP-UX client software, you must manually add the directory to your executable path, even if you accept the default directory.

For most UNIX clients, the executable path is set in the PATH environment variable. Adding the directory containing the Backup executables to your executable path allows execution of Backup commands without entering the full pathname. For example, you would enter nwbackup instead of /opt/nsr/bin/nwbackup.

## Windows NT Clients

On Backup clients for Windows NT, you must manually update and verify certain files, directories, and services, as follows:

- The%SYSTEMROOT%\SYSTEM32\DRIVERS\ETC\HOSTS file must contain the Internet address of the Backup client and the Backup server, unless you are using DNS or Windows Internet Naming Service (WINS). The HOSTS file is a simple

ASCII text file with one line for each Internet Protocol (IP) address. The IP address is the first entry on the line followed by the hostname and all aliases for each computer, for example:

```
127.0.0.1   localhostloopback
137.69.8.1  server  server.companyname.com
137.69.8.2  client  client.companyname.com
```

Your `%SYSTEMROOT%\SYSTEM32\DRIVERS\ETC` directory should contain a sample `HOSTS` file that gives details about adding entries to the `HOSTS` file. Do not delete or comment out the entry for localhost loopback.

If you are using DNS or WINS, verify that the DNS or WINS server has entries for both the Backup client and the Backup server.

- The `SERVERS` file is typically in `C:\WIN32APP\NSR\RES.` Backup uses the contents of this file to control who has the right to request a program to be executed on this client.

  If you want this client to back up to other Backup servers, you must add the names of the additional Backup servers to this file. You can add only one server name per line.

  If you want other clients to be able to perform directed recovers to this client, you will need to add their names to the `\NSR\RES\SERVERS` file. You can add only one client name per line.

  If you want to allow any Backup server to back up this Backup client, delete the `SERVERS` file.

  After you save your changes, you must restart the Backup Remote Exec Service to make your changes take effect.

  To allow any Backup server to back up this Backup client, delete the `SERVERS` file.
- The Backup client for Windows NT must have the latest service pack from Microsoft applied.
- Make sure that the following services are running:
  - Backup Remote Exec Service (`nsrexecd.exe`)
  - Backup Portmapper Service (also known as `portmap.exe`)

  Backup Portmapper Service is an optional service for Backup clients. To enable this service, start it before Backup Remote Exec Service.
- Save sets maintained on an NT DFS (disk file system) link are backed up during a server- or client-initiated backup; however, Backup does not traverse the links or back up the destination files. The Backup server does not permanently modify the last access time for save sets on a DFS link.

## Windows 95 Clients

On Windows 95 clients, you must manually edit and verify certain files, directories, and services, as follows:

- The `HOSTS` file, typically found in `C:\WINDOWS`, must contain the Internet address of the Backup client and the Backup server, unless you are using DNS or WINS. The `HOSTS` file is a simple ASCII text file with one line for each IP address. The IP address is the first entry on the line followed by the host name and all aliases for each computer, as in the following example:

```
127.0.0.1   localhostloopback
137.69.8.1  serverserver.companyname.com
137.69.8.2  client  client.companyname.com
```

- Your Windows 95 directory, typically `C:\WINDOWS`, should contain a sample `HOSTS` file, named `HOST.SAM`, that gives details about adding entries to an actual `HOSTS` file. Do not delete or comment out the entry for localhost loopback.

  If you are using DNS or WINS, verify that this DNS or WINS server has entries for both the Backup client and the Backup server.
- The `SERVERS` file is in `C:\PROGRAM FILES\LEGATO\NSR\RES.` Backup uses the contents of this file to control who has the right to request a program to be executed on this client.

  If you want this client to back up to other Backup servers, you must add the names of the additional Backup servers to this file, one server name per line.

  If you want other clients to be able to perform directed recovers to this client, you will need to add their names to the `\NSR\RES\SERVERS` file, one client name per line.

  If you want to allow any Backup server to back up this Backup client, delete the `SERVERS` file.

  After you save your changes, you must restart the Backup Remote Exec Service to make your changes take effect.

  To allow any Backup server to back up this Backup client, delete the `SERVERS` file.
- The Windows 95 client must have the latest service pack from Microsoft applied.
- Make sure that the Backup Scheduled Backup (`wtcpschd.exe`) is running. Put a copy of Backup Scheduled Backup in the Startup folder to enable scheduled backups to run automatically.

## NetWare Clients

On NetWare clients, you must manually update and verify certain files, directories, and services, as follows:

- The SYS:ETC\HOSTS file must contain the internet address of the Backup client and the Backup server. The HOSTS file is a simple ASCII text file with one line for each IP address. The IP address is the first entry on the line followed by the host name and all aliases for each computer. The HOSTS file should also contain an entry for *localhost*, as in the following example:

```
127.0.0.1   localhostloopback
137.69.8.1  serverserver.companyname.com
137.69.8.2  client  client.companyname.com
```

---

**Caution –** The TCP/IP hostname and the NetWare server name must be identical for the Backup for NetWare client. In the previous example, the NetWare server name replaces the value represented by *client*.

---

- TCP/IP must be loaded and bound correctly in the AUTOEXEC.NCF, for example:

```
load tcpip
load pcntnw board=1 frame=ethernet_ii name=e_ii
bind ip to e_ii addr=137.69.8.2 mask=255.255.255.000
```

  Load and bind TCP/IP before Backup is installed, or some configuration files are not properly updated.

- Other files that affect Backup operation on a TCP/IP network and that are automatically configured during the Backup installation are SYS:ETC\RPCUSERS, SYS:ETC\SERVICES, SYS:ETC\RPC, SYS:ETC\GATEWAYS, SYS:ETC\NET\NETWARE\SERVICES, and RPCNET.CFG, typically found in SYS:NSR.

  Other RPC-based products can also use many of these files, so they might already exist on a client before you install Backup. If the files exist, Backup does not overwrite these files during installation. In most cases, files provided by other RPC-based software work with Backup.

## Macintosh Clients

A Macintosh client must meet the following requirements:

- Macintosh System Software Release 7.1 or later is installed.

- MacTCP Release 2.0.6 or Open Transport is installed.
- MacTCP has Ethernet selected (not EtherTalk).
- Domain Name Services (DNS) are available to the Macintosh. You need to manually configure DNS correctly.

The domain name and the IP address must match the information in the `/etc/resolv.conf` file on the DNS server. The `/etc/resolv.conf` file points your computer to the correct name server. If there is no `resolv.conf` file, the resolver uses the nameserver on the local computer, for example:

```
domain companyname.com nameserver 137.69.8.2
```

- Both the domain name and the fully qualified name of the Backup server must be specified on the Macintosh client, and must appear in the format that follows:

```
domain
domain.companyname.com
```

- The TCP, DNS, and UDP utilities are available for use on the Macintosh. It is advisable to also have the `ping` utility available for troubleshooting.

## Storage Node Affinity

The link between a client resource and a list of storage nodes is called *storage node affinity*. You define storage node affinity in the Storage Nodes attribute in the Clients resource. The default setting for the Storage Nodes attribute on most Backup client resources is the Backup server. For the client resource of a storage node computer, the default setting of the Storage Nodes attribute is the storage node and the Backup server. You can add the names of other storage nodes to the list. The Backup server uses the list in the Storage Nodes attribute to determine which device writes the data from each savestream.

---

**Caution –** When the server's index and the bootstrap save set are backed up, the data is always written to a device that is local to the Backup server. A bootstrap cannot be backed up to a remote device, but a *bootstrap clone* can be written to a remote device. If you use `mmrecov` to recover a bootstrap save set or the server's index, you must recover the data from a local device.

---

During backup, only the devices attached to the storage node computer in the Storage Nodes attribute list are considered to receive that client's data. You *cannot* specify a different list of storage nodes to be used for different operations, but you can add and remove storage node names from the Storage Nodes attribute in the Clients resource at any time.

If a backup fails with the following message, the problem is storage node affinity:

```
no matching devices; check storage nodes, devices or pools
```

Common storage node affinity problems include the following:

- No devices are enabled on the storage nodes in the Storage Nodes list.
- The devices do not have volumes that match the pool required by a backup request.
- All devices are set to read-only.

A common example is when the client has only one storage node in its affinity list and all devices on that storage node are disabled.

You must fix the problem and restart the backup. To fix the problem:

- Enable devices on one of the storage nodes on the client's list.
- Correct the pool restrictions for the devices on the storage node list.
- Add another storage node to the list that has enabled devices meeting the pool restrictions.
- Set one of the devices to read/write.
- Adjust the Save Mount Timeout and Save Lockout attributes for the storage node's Devices resource. For more information, see the online help.

## ▼ Specifying the Level of Backup

You assign each Backup client to an existing schedule in the Schedule attribute of the Clients resource. Backup schedules define what level of backup Backup runs for each calendar day. If none of the existing schedules suit your needs, you can create a custom schedule in the Schedules resource. For more information on schedules, see "Schedule Configuration" on page 41.

## ▼ Specifying When Backup Starts

Two attributes in the Clients resource, Group and Client Priority, determine what time a client's scheduled backup begins.

### Backup Groups

Backup groups determine what time the scheduled backup starts. For each client, in the Clients resource, you select one or more backup groups from the list of available backup groups in the Groups attribute. Use the Groups resource to create custom backup groups. For more information on backup groups, see “Backup Group Configuration” on page 34.

### Client Priority

The Client Priority attribute in the Clients resource specifies the order in which participating clients are probed for the information needed to complete the save set worklist for that client. The Client Priority attribute can contain a value between 1 and 1000. The lower the value, the higher the priority.

The client with the lowest value for the Client Priority attribute is placed at the top of the list to be contacted by the Backup server. If you do not specify a value in the priority attribute, the contact order is random.

While the Client Priority attribute specifies the order of client contact, many variables affect the order in which clients complete their backups, including the following scenarios:

- The backup operation on a client does not begin until the worklists for each of the save sets on the client are complete.
- The amount of work can vary greatly from one client to the next.
- If a client hangs and times out, it is put at the end of the list of clients to be contacted. To increase the number of times each client in a group is retried before the backup attempt is considered unsuccessful, change the value in the Client Retries attribute in the Groups resource.

## ▼ Specifying Which Data Is Backed Up

The Save Set attribute of the Clients resource specifies the data to be backed up for the client computer. You can specify more than one save set in the Clients resource. The Backup server starts a new instance of the client’s save program for each save set you specify.

### What Is a Save Set?

Save sets are groups of files from a single client computer to be backed up by Backup. A save set can include any of the following:

- All the files or filesystems on a client. This is the default condition indicated by the value All
- A filesystem, for example, `/usr`
- A single file, for example `/home/mars/stars.txt`
- A raw space, such as a logical volume
- A database (if you have a BusinesSuite Module installed)

## ▼ How to Use Unique Client/Save Set Combinations

You can use one client license on a computer to back up different portions of its data at different times. This is useful if a client has a large volume of data. To do this you would schedule the client computer backup as several, separate client/save set backups. When you redefine a large filesystem into multiple client/save set instances, you automatically back up a large client file system and balance the system load by avoiding a full backup of the entire file system at one time.

To create several client/save set combinations for a client computer, follow these steps:

1. **Create a client in the Clients resource that specifies a portion of the client's data, for example a single file system, in the Save Set attribute.**

2. **Create another client in the Clients resource that uses the same client host name but specifies a different portion of the client's data in the Save Set attribute.**

   If you specify more than one save set, enter each save set on a separate line.

3. **Associate each client/save set instance with a different backup group to vary the start time of the backups.**

4. **Associate each client/save set instance with a different schedule to specify that each client/save set instance runs its full backup on a different day of the week.**

   You can associate the same save set with more than one client instance, so it can be associated with more than one group or schedule for backup.

   If the default keyword All appears in the Save Set attribute in the Clients resource, all local file systems for the client computer are backed up according to the group and schedule listed in the Clients resource.

   When you configure multiple client resources for the same computer, the most conservative of the assigned browse and retention policies is automatically implemented for all of them.

---

**Caution –** The `core` file is not backed up unless you specify it in the Save Set attribute of the Clients resource.

---

### Logical Volume Backup

A *logical volume* is a type of primary (disk) storage on a client computer that can span several physical disk volumes. The logical volume has its own device address, and it is treated similarly to a disk partition by the filesystem. When Backup backs up data from clients, it has to determine how many save sessions to allocate to each client for best performance. To avoid contention, there should not be more than one backup operation running per physical disk. Backup attempts to allocate different sessions across different physical disks for this reason.

To determine how many save sessions to allocate, the Backup server probes (queries) the clients in a backup group (using the `savefs -p` command) to find out what data to back up and where the data is physically located. Backup tries to determine whether there are logical volumes. It stores this information in two variables, *disk-number* and *maximum-sessions*, according to the following rules:

- When the group of volumes or disks that contain logical volumes is not part of the device path, all logical volumes on the client computer are assigned to the same *disk-number,* and *maximum-sessions* is set to the number of logical volumes on the client computer.
- When the group of volumes or disks that contain logical volumes is part of the device path, all logical volumes *within the volume group* are assigned to the same *disk-number,* and *maximum-sessions* is set to the number of logical volumes within the volume group.

The server uses the output from the `savefs` probe to allocate its save sessions (up to the maximum server parallelism) across the clients in the backup group:

1. First, the server allocates one save session per client in the backup group.
2. Then, if there are still save sessions available, it allocates one save session per physical disk on each client.
3. If, after that, there are still save sessions available, it allocates save sessions to each *disk-number* value, up to the limits in *maximum-sessions* for each client and client parallelism.

## ▼ Specifying the Longevity of Backup Data

Use browse and retention policies to specify how long data is available for recovery. You can specify browse and retention policies for each client.

## What Are Browse and Retention Policies?

The Backup server maintains one file index for each client computer (regardless of the number of client resources configured for it) and one media database that tracks data from all clients. Each time a backup is completed, Backup creates entries for the backed-up files in the client file indexes. The media database stores one entry for each save set and storage volume during each backup operation.

Each client file index is a browsable structure of data from a single client computer. Users can specify anything from a single file to a complete filesystem and direct Backup to reconstruct the data during a recover session to look exactly as it did at a specific time. The information that the client index contains and coordinates enables Backup to automatically handle situations such as assembling data from backups based on levels, and to accommodate all file or directory renamings or deletions. Backup uses browse policies to manage the life cycle of data and to automatically control the size of the client file index.

The *browse policy* determines how long files are maintained in the client's file index on the Backup server. During the period of the browse policy, users can browse backed-up data in the Backup recover program (`nwrecover`) and select individual files or entire filesystems for recovery. After the browse policy for a file is exceeded, Backup automatically deletes the entry for that file. Backup deletes these entries to manage the size of the client index, which can grow rapidly: one entry for each file backed up during each scheduled backup of the client.

The media database is the structure that tracks the location of save sets on storage volumes. Backup uses a *retention policy* to manage the longevity of Backup managed data. Data is recoverable as long as entries exist in the media database; there is nothing to be gained by rushing to delete media database entries. For all these reasons, the media database retention policy *does not* trigger the automatic removal of media database entries. Instead, the retention policy determines how long an entry for a save set remains protected from being accidentally written over.

The retention policy determines how long save sets are maintained in the Backup server's media database. For at least the period of the retention policy, you can recover a client's backed-up save sets from media. No save set is considered recyclable until, at a minimum, it has exceeded its retention policy. No storage volume can be relabeled and written over until, at a minimum, all save sets on the storage volume have exceeded their retention policies. Theoretically, entries for a save set or a storage volume can remain in the media database forever, long after the retention policy has been exceeded. Entries are removed from the media database only if a storage volume is relabeled or if you manually delete the entries.

## How the Browse Policy Works

You can recover a file that has an entry in the client file index through the Backup recover program, which enables users to browse and mark files and initiate data recovery. Client file index entries are not necessarily deleted the same day that the browse policy is exceeded. Backup does not remove the entry for a file until *all* the save sets that are dependent on the file have also exceeded their browse policies. In general, the entries for a full backup that are older than the browse policy are not removed until an additional length of time equal to one backup cycle passes as well. This extra time ensures that you can reconstruct a file to any point in time included in the browse policy period.

The following figures demonstrate how a browse policy affects data availability in the client file index. For more information about schedules, see “Schedule Configuration” on page 41, and for more information about backup levels, see “Backup Levels” on page 45.

In the following figure, both the backup cycle and the browse policy are set at 1 week. A backup cycle is the length of time between full backups. Entries for the first full backup on October 2 remain in the client file index until all the incremental and level 5 backups that depend on it exceed the one-week browse policy. The full backup performed on October 2 is not removed until October 16, when the incrementals and level 5 that depend on the full backup expire.

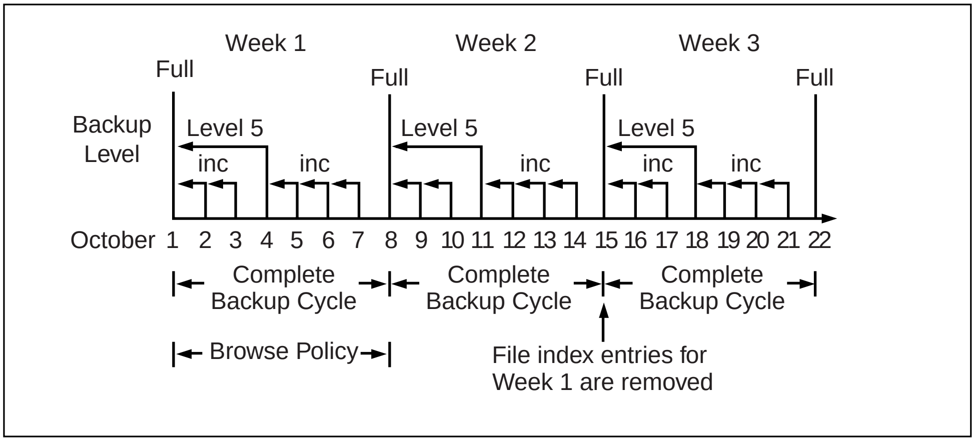


**FIGURE 5-1** One-Week Browse Policy

To illustrate why the browse policy works this way, suppose that on October 12, you decide that you want to recover information backed up on October 6. The backup performed on the 5th is an incremental backup dependent on the October 5 backup, which is a level 5 backup. The October 5 level 5 backup, in turn, is dependent on the full backup performed on October 2. The entry for the full backup performed on October 2 must remain in the client file index for a period of time equal to the browse policy (one week) plus one complete backup cycle (one additional week)—that is, until the level 5 backup on October 5 and all incremental backups dependent on the full backup pass their browse policy. In the example shown in FIGURE 5-1 on page 113, entries from the Week 1 backup cycle are removed from the client file index on October 16.

In the following figure, the browse policy is 2 weeks, which is twice as long as the backup cycle (1 week). In this example, on October 19 a user can still find browsable entries in the client file index from backups created on October 5. The backup performed on October 6 is an incremental backup dependent on the October 5 backup, which is a level 5 backup. The October 5 level 5 backup, in turn, is dependent on the full backup performed on October 2. The full backup performed on October 2, and the incremental and level backups that depend on it, must remain in the client file index for a period of time equal to the browse policy (two weeks), plus one complete backup cycle (one additional week). In this example, entries for the Week 1 backup cycle are not removed from the client index until October 23.

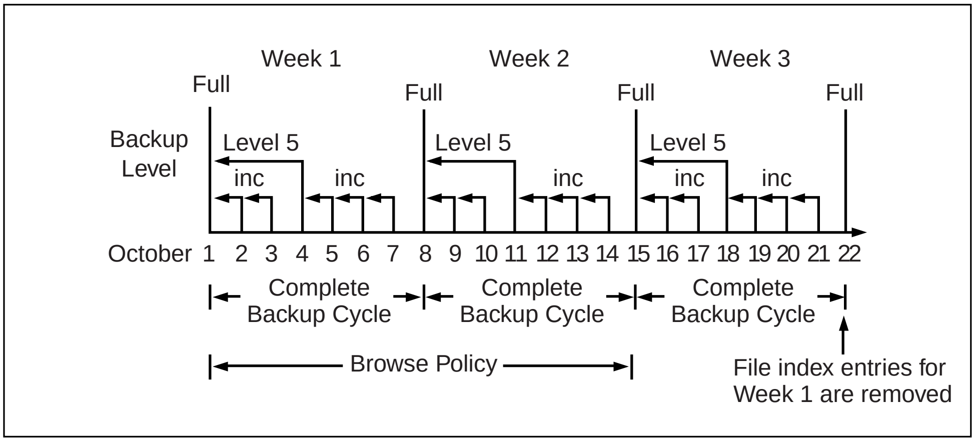


**FIGURE 5-2** Two-Week Browse Policy

## How the Retention Policy Works

The Backup media retention policy specifies a period of time during which backed-up data is protected from accidental overwrite. After the retention period is exceeded, the save set is eligible to change its status from recoverable to recyclable. The save set’s status, however, does not change to recyclable until it and all the save sets that depend on it have passed their retention policy. Backup keeps track of save set dependencies regardless of whether the dependent save sets are stored on the same media volume or on different volumes. The expiration of a save set’s retention policy does not remove the save set’s entries from the media database.

When the retention policy for *every* save set on a volume expires *and* the status for every save set on a volume changes from recoverable to recyclable, Backup changes the mode of that storage volume to recyclable. Since a volume can contain save sets from multiple backup sessions, all with different retention policies, the mode of a volume might not change to recyclable for a long time. The term recyclable is best understood as “eligible for recycling.”All the data on the volume remains available for recovery using either save set `recover` or the `scanner` command. All the entries for recyclable save sets remain in the media database.

The change in status to recyclable is a passive reminder that you can overwrite the volume if conditions are right. If you place the volume in an autochanger or mount the volume in a stand-alone device and enable the auto media management attribute in the Devices resource, the volume is available for relabel and use by Backup. The existing data is nonrecoverable after the volume is relabeled, so the entries for the overwritten save sets are removed from the media database. For more details about this feature of auto media management, see “How Backup Selects a Storage Volume for Relabeling” on page 89.

The save set’s entries are also removed from the media database when you manually delete a volume from the Backup volume inventory. However, the data on a volume that you delete manually is still available for recovery using the `scanner` program. The `scanner` program retrieves the information needed to re-create entries in either the client file index, in the media database, or in both places. If you re-create the entries in the client file index, a user with the proper permissions can recover data through the Backup recover program (`nwrecover`). If you re-create the save set’s entries in the media database, a user with Backup administration privileges can recover data with save set `recover`. See Appendix B, “Command Line Reference Utilities,” or refer to the `scanner` man page for more information on how to use the `scanner` program.

FIGURE 5-3 on page 116 illustrates how a retention policy works. In this example, the backup cycle is set at 1 week and the retention policy is set at 3 weeks.

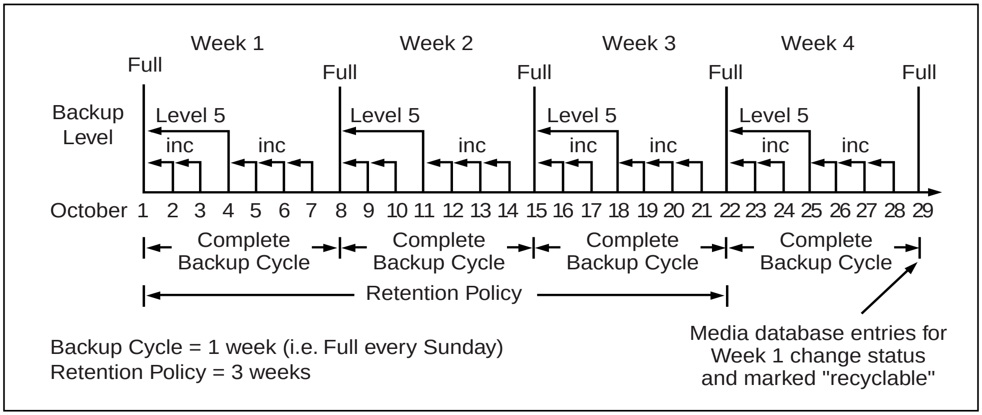

Week 1
Week 2
Week 3
Week 4
Full
Full
Full
Full
Full
Backup Level
Level 5
Level 5
Level 5
Level 5
inc
inc
inc
inc
inc
inc
inc
October 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
Complete Backup Cycle
Complete Backup Cycle
Complete Backup Cycle
Complete Backup Cycle
Retention Policy
Backup Cycle = 1 week (i.e. Full every Sunday)
Retention Policy = 3 weeks
Media database entries for Week 1 change status and marked "recyclable"

**FIGURE 5-3** One-Week Backup Cycle; Three-Week Retention Policy

The save set entries for Week 1 have passed their browse policy and retention policy, but they remain available for recovery using the `scanner` program until you relabel the volume. When all the save set entries on a volume change status to recyclable, the volume mode changes from full or appendable to recyclable, and the volume is ready to be relabeled for reuse. For more information about storage volume modes, see Table 4-3 on page 92.

---

**Caution –** Once you relabel a volume, the data on the volume cannot be recovered.

---

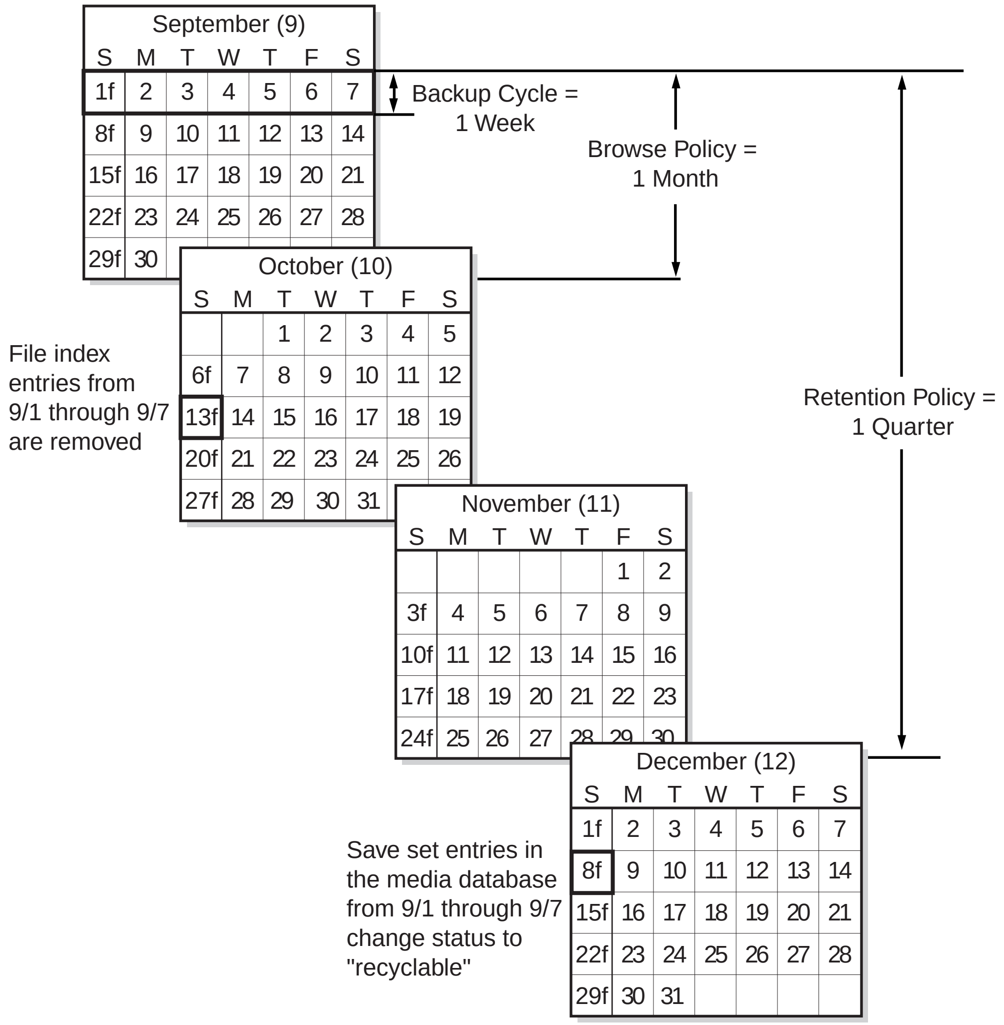
For more information about schedules, see “Schedule Configuration” on page 41, and for more information about backup levels, see “Backup Levels” on page 45.

## How the Browse and Retention Policies Manage the Data Life Cycle

The browse and retention policies that you associate with a client save set control both the growth of the client file index and the media database, and how long data remains available for recovery. FIGURE 5-4 on page 118 traces the data life cycle through the client file index and the media database. In the example, the entries for the September 1 through September 7 backup cycle remain in the client index for 1 month (the browse policy), plus the length of a full backup cycle (1 week), to ensure that all dependent entries pass their browse policies. In this case, the file index entries for the September 1 through September 7 backup cycle are removed on October 13. Since the entries exist in the client file index, you can browse and recover the data through the `recover` program’s GUI (`nwrecover`). As long as the save set’s file entries remain in the client file index, the status of the source save sets is browsable. After the save set status changes from browsable to recoverable, you cannot perform file recovery directly.

The status for each save set backed up during the September 1 through September 7 backup cycle remains recoverable until their retention policies expire, *plus* however long it takes for all the dependent save sets to pass *their* retention policies. In this case, the entries from the September 1 through September 7 backup cycle change from recoverable to recyclable on December 8. When all of the save set entries on a volume change status to recyclable, the mode of the volume itself changes from either full or appendable to recyclable.

While the status of a save set is either recoverable or recyclable, you can recover any save set from the storage volume by using either the save set recover operation or the `scanner` program. Alternatively, you can use the `scanner` program to re-create a save set’s entries in the client file index, which enables file recovery directly from the GUI. For more information about using Save Set Recover and the `scanner` program, see “Save Set Recover and the `scanner` Program” on page 120.



September (9)

| S | M | T | W | T | F | S |
|---|---|---|---|---|---|---|
| 1f | 2 | 3 | 4 | 5 | 6 | 7 |
| 8f | 9 | 10 | 11 | 12 | 13 | 14 |
| 15f | 16 | 17 | 18 | 19 | 20 | 21 |
| 22f | 23 | 24 | 25 | 26 | 27 | 28 |
| 29f | 30 | | | | | |

October (10)

| S | M | T | W | T | F | S |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| 6f | 7 | 8 | 9 | 10 | 11 | 12 |
| 13f | 14 | 15 | 16 | 17 | 18 | 19 |
| 20f | 21 | 22 | 23 | 24 | 25 | 26 |
| 27f | 28 | 29 | 30 | 31 | | |

November (11)

| S | M | T | W | T | F | S |
|---|---|---|---|---|---|---|
| | | | | | 1 | 2 |
| 3f | 4 | 5 | 6 | 7 | 8 | 9 |
| 10f | 11 | 12 | 13 | 14 | 15 | 16 |
| 17f | 18 | 19 | 20 | 21 | 22 | 23 |
| 24f | 25 | 26 | 27 | 28 | 29 | 30 |

December (12)

| S | M | T | W | T | F | S |
|---|---|---|---|---|---|---|
| 1f | 2 | 3 | 4 | 5 | 6 | 7 |
| 8f | 9 | 10 | 11 | 12 | 13 | 14 |
| 15f | 16 | 17 | 18 | 19 | 20 | 21 |
| 22f | 23 | 24 | 25 | 26 | 27 | 28 |
| 29f | 30 | 31 | | | | |

**FIGURE 5-4** Data Life Cycle in the Client Index and the Media Database

On October 13, all data entries from September 1 to September 7 are removed from the client file index. On December 8, the save set entries from September 1 to September 7 in the media database change status from recoverable to recyclable. After all save sets on a volume change status from recoverable to recyclable, the volume mode changes to recyclable. If auto media management is enabled, the volume may be relabeled automatically by Backup to satisfy a volume mount request. After you relabel the volume, all existing data on the volume is unavailable for recovery.

---

**Caution –** When you relabel a volume for reuse within the same pool, the volume identification (the volume name as it appears on the volume label) remains unchanged. Even so, after relabeling, the information that Backup needs to locate and access all existing data on the volume is destroyed and neither the Save Set Recover feature nor the `scanner` program are options. At this point, the volume is ready for new data. All existing data is inaccessible and is overwritten.

---

## Save Set Status Values

Backup assigns to each backed-up save set a status based on the success of the backup or the age of the save set data. The save set status changes in the following situations:

- When the save set exceeds its browse policy. For more information about browse policy, see "How the Browse Policy Works" on page 113.
- When the save set exceeds its retention policy and all save sets dependent on the save set also exceed their retention policies. For more information about retention policy, see "How the Retention Policy Works" on page 114.
- When you manually change the save set status.

Table 5-1 provides a list of all the possible values for save set status.

**TABLE 5-1** Save Set Status Values

| Status Value | Meaning | Description |
|---|---|---|
| `abort` | Aborted | You aborted the backup for this save set manually or a crash occurred during the operation. This save set is considered immediately eligible for recycling. |
| `brows` | Browsable | The files in this save set retain entries in the client file index. You can restore all the files using an index-based recover. |
| `inpro` | In progress | This save set is currently being backed up. |

**TABLE 5-1** Save Set Status Values *(Continued)*

| Status Value | Meaning | Description |
|---|---|---|
| `recov` | Recoverable | The files in this save set do not have browsable entries in the client file index and have not passed the retention policy. |
| `recyc` | Recyclable | The save set and all the save sets that are dependent on this save set for recovery have exceeded their retention policies. |
| `scann` | Scanned-in | The client file index entry for this save set was restored with the `scanner` program. This entry remains in the client file index and media database until you remove it manually. |
| `susp` | Suspect | An attempt to recover this save set failed. The `recover` program could not read all the blocks of the save set, for example, if there was a bad spot in the tape. |

## Customizing Policies

Use the Policies resource to create a custom browse policy or retention policy. In the Policies resource, give the policy a unique name and specify a time period. After you define a policy, it is available as a choice in the Browse Policy and Retention Policy attributes in the Clients resource.

# Save Set Recover and the `scanner` Program

Use the save set recover feature to recover backed-up data that has passed the period of its browse policy but is still in the media database. You can initiate the save set recover feature either from the command line by executing the `recover` program and providing specific save set identification numbers (ssid) as options, or from the Backup administration program (`nwadmin`). You can specify individual files or directories by including the exact path along with the ssid. Permission to perform a save set recovery is granted only to root.

Use the save set recover feature *only* when the entries have been removed from the online file index (when the save set has passed its browse policy). When you perform a save set recover operation, you must recover the level full backup first, then recover the other backups in level order from 1 through 9, then recover the incremental backups.

If no entries for the volume exist in the media database, use the `scanner` program to re-create client file index entries or re-create media database entries. The `scanner` program can read the storage volume directly, without assistance from Backup.

To find the volume that contains the file you want, use the `mminfo` program if the volume is still in the media database or the `scanner` program if the volume is no longer in the media database. The `mminfo` and the `scanner` programs provide detailed information of the contents of the volume. This information includes:

- Name of the backup volume
- Name of the save set that contains the file you want
- Name of the client to which the file belongs
- Date and time the file was backed up

## ▼ How to Rebuild a Save Set Entry in the Client File Index

If the file is not browsable (which means that the save set's browse policy has expired) but its save set is still tracked by Backup in the media database (which means that the save set's retention policy has not expired), follow these steps to recover the save set's entry back into the client file index:

1. **Run the** `mminfo` **program:**

```
# mminfo -a -v volume-name
```

2. **From the mminfo output, find the SSID that you believe contains the file you want.** *Make sure it is not the bootstrap SSID.*

3. **After you have the proper SSID, replace the save set entry in the file index with the** `scanner` **program:**

```
# scanner -i -S save-set-id  device-name
```

4. **Use the Backup** `recover` **program to mark the file for recovery.**

---

**Caution –** If the save set spans volume boundaries, use the `scanner` program to read from *all* the volumes. Otherwise, the client file index is not fully rebuilt, making it impossible to perform an online recovery of the files in this save set.

---

5. **Use the Backup** `recover` **program to mark the file for recovery.**

If the save set that contains the file is not browsable and the save set is not represented in the media database, both the browse and retention policies have expired. Follow these steps to rebuild the save set's entry in both the client file index and the media database:

1. **Run the** `scanner` **program on the backup volume that you believe contains the file you want (make a guess based on the adhesive label on the volume or use the procedures listed under "How to Find a Volume Name" on page 89:**

```
# scanner device-name
```

2. **Use the output from the scanner program to decide whether to reintroduce the contents of this volume into the client file indexes and whether the save set you want to rebuild is on this volume. You must locate** *all* **the volumes that contain this save set ID.**

3. **After you have determined which volumes to reintroduce into the online indexes, run the scanner command:**

```
# scanner -i device-name
```

   The `scanner` command prompts for a new volume until you terminate it. To rebuild the indexes completely, you must scan in *all* the volumes that contain the SSID.

4. **Use the** `nwrecover` **program to browse the file index for the file you want to recover.**

## ▼ How to Recover an Entire Save Set to the Backup Server

To recover an entire save set directly to your disk volume, use the following options to invoke the `scanner` program:

```
# scanner -S save-set-id device-name | uasm -rv
```

This command reads all the information associated with the SSID from the volume and places a copy of this data on the Backup server in the exact way that it is stored on the backup volume. In other words, the backup volume may contain files for a client, but is recovered to the Backup server's hard drive.

If you want to be sure this action is correct before you perform it, add the **–n** flag to the `uasm` command. The `–n` flag sends the output from `scanner` to `/dev/null` and lists all the filenames contained in the save set.

You could also use `rsh` (or its equivalent) with the following command to recover the save set to the client, if the save set originated on a Backup client instead of the Backup server:

```
# scanner -S ssid device-name | rsh client "(cd destdir; /path/uasm -rv)"
```

## ▼ How to Recover One File Directly From a Volume

To recover a single file from a volume, run one of the following commands:

```
# scanner -S save-set-ID device-name | uasm -rv filename
```

or

```
# scanner -S save-set-ID  device-name | uasm -rv -m source=dest filename
```

The –m option of uasm maps (relocates) the recovered file from the *source* to the *dest* *(*destination) directory.

## ▼ How to Add Processing Instructions to the Scheduled Backup

The Directive and Backup Command attributes in the Clients resource add instructions for client-side data processing to a scheduled backup.

### What Are Directives?

Directives are special programs Backup applies to save set data to initiate additional data processing. For example, a compression directive can reduce the amount of data you back up, possibly even eliminating the need to change backup volumes on the days you perform a full backup. Directives appear as selectable options associated with the Directives attribute in the Clients resource.

A directive contains instructions to assist the backup process, maximize the efficiency of a backup, and handle special files. During a scheduled backup of a client save set, Backup applies directive instructions to specified files. For example, you can use the null directive to omit certain files from the backup entirely.

## ▼ How to Create Customized Directives

Use the Directives resource to create a directive and apply it to a specific client through the Clients resource. Because every environment is different, it is impossible to prescribe directive-writing rules that work in every case. Instead, examples of the most commonly requested customizations are provided below to give you models to follow.

## Example: Directives to Back Up Specific Directories on a UNIX Client

Assume you want to save only the directory /aaa/zzz on client 123 and no others. This directive restricts Backup from walking the directories /aaa and /zzz to locate and back up other files or subdirectories. This directive invokes a UNIX application-specific module (uasm) called null. The use of null skips files in the directory specified; the use of +null skips files in the directory specified as well as those below the directory specified. The content of the directive to back up only /aaa/zzz appears as follows:

```
<< / >>
    uasm: aaa
    null: * .?*
<< /aaa >>
    uasm: zzz
<< /aaa/zzz >>
    +uasm: * .?*
```

In another example, assume you want to back up all non-root mounted disks, and back up the /home and /users directories off the root disk. You also want to back up the cron files and the calendar databases. For each client, the Save Set attribute contains the value All. The directive appears as follows:

```
<< / >>
    uasm: home users var
    null: * .?*
    +null: core
<< /home >>
    +compression: * .?*
    +null: core
<< /users >> *
    +compression
    +null: core
<< /var >>
    uasm: spool
    null: * .?*
    +null: core
<< /var/spool
    uasm: calendar cron
    null: * .?*
    +null: core
<< var/spool/calendar >>
    +compression: * .?*
    +null: core
<< var/spool/cron >>
    +compression: * .?*
    +null: core
<< /cdrom >>
    null: * .?*
<< /opt >>
    null: * .?*
<< /tmp >>
    null: * .?*
<< /usr >>
    null: * .?*
```

The use of null as part of a directive instructs Backup not to save the specified files during the particular backup, but to include an entry in the index listing created by Backup to indicate that the files were included in the backup operation. Because the files are included in the index, the filenames are available for browsing in the directory, and the view of the filesystem through Backup corresponds to the actual filesystem. That is, the recover program’s GUI displays files that are available for recovery, even if you skipped the files in more recent backups and the data available for recovery is not as recent as the data available for other files.

This behavior differs slightly from the behavior of the skip uasm. The skip uasm results in a view of your filesystem from the browser that reflects the backed-up data, not the closest approximation to the actual filesystem. For these reasons as well as other, more technical, advantages, the use of null is recommended over skip.

---

**Caution –** In all instances where a wildcard character appears, there must be a space inserted between the wildcard and the next character, to avoid unexpected results. For example, if you implemented the previous directive example that used the wildcard * .?* to indicate that the directive should be implemented upon all files and accidentally omitted the space between the asterisk (*) and period (.), the actual result would be that only files with a period (.) character in the middle of their file name would be subject to the directive. In this instance, the intent was to have the directive act on all files: the asterisk (*) indicates all files, while the dot (.) indicates that the directive should also act upon all hidden files.

---

## Backup Command

You can customize your client backups by creating additional programs that affect the way Backup backs up client filesystem data. For example, you can create a program that shuts down either a mail server or a database before Backup performs a backup operation and then restarts the mail server or database after the backup is completed. Or you can create a program that prints a message (such as `backup started at 3:33 a.m.`) before the backup operation begins, then executes the backup on the client data and prints a message when the backup is completed (such as `backup completed at 6:30 a.m.`).

You can customize a client's scheduled backups through two different methods, as follows:

a. **Create a script that invokes the** save **program as part of its instructions. You enter the name of your customized script in the Backup Command attribute in the Client resource for a given client computer. When the client is backed up, the customized script is invoked rather than the client's standard** save **program. The instructions in the customized script are run separately for each save set backed up for the client. See "How to Use** save **in a Customized Backup Script" on page 129 for further information.**

b. **Configure the client's backup to use the** savepnpc **program instead of the standard** save **program. To configure a client to run** savepnpc**, enter** savepnpc **in the Backup Command attribute in the Client resource for the client computer. When the client is backed up, the** savepnpc **program is invoked rather than the client's standard** save **program. The** savepnpc **program is**

**invoked once during the client’s backup session, rather than once for each save set. See “How to Use the savepnpc Command” on page 127 for further information.**

Consider the following issues as you determine what level of customization works best for your environment:

- Amount of disk space available
- Whether you have client data that doesn’t need to be backed up every time (for example, company email)
- Whether you want Backup to send special messages (in addition to the savegroup completion reports) about the backups it executes

## How to Use the savepnpc Command

You can run pre-processing and post-processing commands that execute only once per client backup by entering savepnpc in the Backup Command attribute. The savepnpc program, like the save program, saves files to long-term storage.

The default implementation of the savepnpc program is to implement a timeout condition you can set to indicate when the post-processing commands need to be run without waiting for all the save sets to be backed up. Specify the timeout condition in a format that nsr_getdate(3) can understand.

The first time that a backup group with a client that uses the savepnpc program runs, a standardized /nsr/res/<group-name>.res file is created, where the value of group-name is the same as the name in the Group resource selected for the client. If the client belongs to multiple backup groups, a separate /nsr/res/<group-name>.res file is created when the client is backed up as part of the other groups. The /nsr/res/<group-name>.res file that is created initially contains a default set of pre-processing, post-processing, and timeout commands to perform:

```
type: savepnpc;
precmd: “echo hello”;
pstcmd: “echo bye”, “/bin/sleep 5”;
timeout: “12:00pm”;
```

You can use your favorite text editor to customize the file to include other instructions for the next time the client is backed up. You can also make copies of the file to customize for other backup groups, *as long as* the filename you copy to is in the format <group-name>.res and the copied file remains in the /nsr/res directory on the Backup server.

Before performing the first save operation on a client, savepnpc performs any pre-processing commands that exist in the /nsr/res/<group-name>.res file. After the last save operation is successfully completed on the client, savepnpc performs any post processing commands listed in the /nsr/res/<group_name>.res file.

All of the messages generated by savepnpc are written to the /nsr/logs/savepnpc.log file on the client. The format of the log entry is similar to the following example:

```
09/03/98 19:29:01 preclntsave: On mars:
Ran pre-processing cmd(s) successfully.
09/03/98 19:29:02 preclntsave: On mars:
Successfully spawned off pstclntsave.
09/03/98 19:29:07 pstclntsave: On mars:
All savesets on the worklist are done.
09/03/98 19:29:13 pstclntsave: On mars:
Ran all post-processing cmd(s) successfully.
```

The corresponding entries written to the syslog are similar to the following example:

```
Sep  8 03:33:01 jupiter syslog: Backup Savegroup: (info) starting
default_savepnpc (with 1 client(s))
Sep  8 03:33:12 jupiter syslog: Backup Savegroup: (notice)
default_savepnpc completed, 1 client(s) (All Succeeded)
Sep  8 03:33:12 jupiter syslog: Start time:   Tue Sep  8 03:33:01
1998
Sep  8 03:33:12 jupiter syslog: End time:     Tue Sep  8 03:33:12
1998
Sep  8 03:33:12 jupiter syslog:
Sep  8 03:33:12 jupiter syslog: --- Successful Save Sets ---
Sep  8 03:33:12 jupiter syslog:
Sep  8 03:33:12 jupiter syslog: * mars:/testfile hello
Sep  8 03:33:12 jupiter syslog:   mars: /testfile
level=incr,      0 KB 00:00:01      0 files
Sep  8 03:33:12 jupiter syslog:   mars: /space/nsr/index/mars
level=9,         0 KB 00:00:01      0 files
Sep  8 03:33:12 jupiter syslog:   jupiter: bootstrap
level=9,        19 KB 00:00:02      6 files
Sep  8 03:33:12 jupiter syslog:
Sep  8 03:33:14 jupiter syslog: Backup index: (notice) nsrim has
finished checking the media db
```

If you invoke the savepnpc command on a client without setting up a /nsr/res/<group_name>.res file in advance, the default commands are executed, and a file with the name of the group that the client backed up with is created for subsequent instances. If the client was backed up under more than one group, <group-name>.res files are created in /nsr/res for those groups as well. You can edit the file to modify the command invoked by the precmd or pstcmd attribute, or you can modify the value for the timeout attribute. To add more than one command sequence to the `precmd` or `pstcmd` attributes, use a comma (,) to separate the commands. The command line for each attribute must end with a semicolon (;).

See "`savepnpc`" on page 331 of Appendix B and its man pages for more detailed information about the command line options for `savepnpc`.

## ▼ How to Use `save` in a Customized Backup Script

You can enter the name of a customized script in the Backup Command attribute in the Clients resource that includes additional processing instructions. If an entry exists in the Backup Command attribute, the script associated with it is executed instead of the default `save` program when scheduled backups are initiated.

The instructions you include in your script are performed for each save set that is defined for the client, rather than on a per-client basis. If you specify a save set value of All, your script is executed the same number of times as the number of filesystems on the client. The simplest implementation of a customized backup command is to create a special separate client with a single save set listed in the Save Set attribute.

---

**Caution –** Unlike `savepnpc`, a new instance of the customized script, whose name you enter in the Backup Command attribute, is invoked for each save set listed in the Save Set attribute, just like the standard `save` program. Bear this in mind when you create a Client resource with a customized Backup Command for a database, because a command to shut down the database is executed for each save set that you listed.

---

The syntax you use to create the backup program or batch file must adhere to the criteria described in the following list. The list is detailed and includes programming details. *Do not attempt to write your own backup command unless you can follow these recommendations.*

- The backup program name must begin with either the prefix `save` or `nsr` and cannot exceed 64 characters.
- The backup program must reside in the same directory as the Backup `save` command.
- The Backup `save` command must be used in the backup program to ensure that the data is properly backed up.
- All commands within the program file must be successfully executed; otherwise, Backup cannot complete the remaining instructions.
- When you invoke the Backup `save` command, invoke the command with the following arguments: `save "$@"`. Doing so enables the `save` command in your batch file to accept the arguments usually passed to it by the Backup `savefs` program during a routine backup operation.

The program commands should be placed in the following order:

1. Run a pre-processing command before a client backup (optional).
2. Back up the data using the Backup `save` command (mandatory).
3. Run a post-processing command after a client backup (optional).

Follow these steps to create a pre- or post-backup command:

1. **Use a text editor to create a program file in the directory where the Backup save command resides.**
2. **Enter the name of the backup program in the Backup Command attribute of the Clients resource.**

   Try backing up the client to ensure that the backup command you created works.

   The following script is an example of a custom script that does pre- and post-processing. This script locks a ClearCase version object base (VOB), does the backup, then unlocks the VOB.

```
#!/bin/sh
# export the SHELL that we are going to use
SHELL=/bin/sh
export SHELL
# export the correct PATH so that all the required binaries can be
found
case $0 in
/* ) PATH=/usr/atria/bin:/bin:/usr/bin:`/bin/dirname $0`
c=`/bin/basename $0`
;;
* )PATH=/usr/atria/bin:/bin:/usr/bin:/usr/sbin
c=$0
;;
esac
export PATH
# These are the valid statuses which save reports on completion of
the backup
statuses=”
failed.
abandoned.
succeeded.
```

```
completed savetime=
“
# Perform the PRECMD (Lock VOB)
/usr/atria/bin/cleartool setview -exec “/usr/atria/bin/
cleartoollock -c \
  ‘VOB backups in progress’ -vob /cm_data/mis_dev” magic_view >
/tmp/voblock.log 2>&1
# Perform backup on client
save “$@” > /tmp/saveout$$ 2>&1
# cat out the save output
cat /tmp/saveout$$
# search for the backup status in the output reported by save
for i in ${statuses}; do
      result=`grep “${i}” /tmp/saveout$$`
      if [$? != 0]; then
               echo ${result}
      fi
done
# Perform the POSTCMD (Unlock VOB)
/usr/atria/bin/cleartool setview -exec “/usr/atria/bin/
cleartoolunlock -vob
/cm_data/mis_dev” \
    magic_view > /tmp/vobunlock.log 2>&
# make sure to gracefully exit out of this shell script
exit 0
```

## Allowing Remote Access Rights to Other Clients

Backup clients are preconfigured so that only the client itself can browse or recover its own files. If your company is concerned about security, leave the Remote Access attribute blank, so that only the client itself can recover its backed-up files.

To give other users or computers permission to recover a client’s files, enter the user ID and host name (in the format *user@hostname*) or netgroup name (if you are using NIS) in the Remote Access attribute in the Clients resource.

When you enable remote access rights, authorized users can view and recover files from other Backup clients from the Backup `recover` program. In the `recover` program, change to the client, then browse or recover the files you want.

To restrict permission to execute backup commands (`save` and `savefs`) on a client during a scheduled backup, enter a user ID in the Remote User attribute in the Clients resource. When this attribute is blank, the user name, by default, is `root`.

# Modifying or Deleting an Existing Backup Client

You can change the settings for existing clients or delete their configuration from the Backup server.

To change the attributes of an existing client, display the Clients resource, select the client you want to modify, change the values in the client's attributes, then apply your changes.

To delete an existing client, display the Clients resource, select the client you want to delete, then select Delete.

# How Backup Enforces Client Licensing

When a backup begins, a Backup client passes its attributes to the Backup server. The Backup server uses this information to verify that the client is allowed to back up to it. The following read-only attributes in the Clients resource are used for client licensing enforcement:

- Client OS type
- CPUs
- Backup client software version
- Enabler in use

When you enable a Backup server, its enabler allows a specific number of clients for its platform. For example, a Backup server on Solaris is licensed for a specific number of Solaris or SunOS clients. If you want to connect clients of other platforms to the Backup server, you must purchase a ClientPak enabler for those platforms.

You can define more clients than the number of clients allowed by the enabler codes stored in the server. The Backup server enforces the number and type of clients at backup time when it verifies that the number of connected clients is less than or equal to the number of clients allowed by its enabler codes, and the clients are of the types allowed by its enabler codes.

There are two types of client licenses: server and workstation. The type of client license a computer uses when it backs up is determined by its operating system. Computers of the following platforms *must* back up with a server client license:

- Any version of UNIX that Backup supports
- Windows NT Server
- NetWare

Computers of other platforms, for example, Windows 95, Windows NT Workstation, OS/2, and Macintosh, can back up with workstation client licenses. If computers that can back up with workstation licenses are waiting to back up and only a server license is available, the backup can proceed because a computer with a workstation license can borrow a server license. But a computer that requires a server license cannot borrow a workstation license. The following tablelists client types supported by Backup.

**TABLE 5-2** Client Types Supported by Backup Enablers

| Enabler | Client Types |
|---|---|
| Windows NT server | Windows NT Server, Windows NT Workstation, Windows 95, and Windows 3.1x |
| UNIX ClientPak | AIX, DIGITAL UNIX, HP-UX, IRIX, Sequent Dynix/ptx, Solaris, SunOS, SCO, UnixWare |
| ClientPak for Windows NT | Windows NT Server, Windows NT Workstation, and Windows 98/95 |
| ClientPak for Macintosh | Macintosh |
| ClientPak for HP-MPE | HP-MPE |
| ClientPak for NetWare | NetWare |
| ClientPak for PC Desktops | Windows NT Workstation and Windows 98/95 |

In the following cases, backups are rejected because of licensing enforcement with the corresponding error messages:

- If the client requires a server license and there are no more available server licenses allowed by the enablers, the backup is rejected with the following error:

```
RAP error, Too many clients. Maximum is 2
```

- If the client type is not allowed by the enabler code and a ClientPak enabler is not present, the backup is rejected. An example is a server with a Solaris server enabler and the client is an HP-UX client, as in the following error message:

```
RAP error, ClientPak for UNIX does not exist in the system.
```

- If no more licenses are available for that type of client and no server licenses are available to borrow, the backup is rejected with the following error:

```
RAP error, Too many clients. Maximum is 2
```

Backup clients released before Backup 5.0 for UNIX identify themselves to the Backup server at backup time. The Backup server uses a workstation license to back up these clients.

Enablers of Backup products released before Backup 5.0 for UNIX are considered server licenses.

# Backup and Recovery With Backup Client Applications

Backup provides client applications for backup and recovery functions. Users on client computers can initiate backups and recoveries using these programs.

## Manual Backup from the Backup Client Application

At the command line, enter `nwbackup` to start the Backup program. This program is a client-side program that runs manual backups. Manual backups are intended for quick backups of a few files initiated by a user. They do not generate bootstrap files or back up indexes, so they cannot replace scheduled backups.

In the Backup program, select the files to back up, then start the backup. Backup saves the selected files to backup volumes and makes entries in the client's file index and media database. During the next scheduled backup, the revised index and media database are backed up.

For specific instructions on using the GUI version of the Backup program (`nwbackup`), refer to the online help.

## How to Start a Recovery From the Backup Client Application

At the command line, enter `nwrecover` to start the Backup recovery program. In this program, users can browse backed-up files and filesystems from their client computer and recover files as they were at a specific point in time. The versions of files and filesystems available in the Backup recovery program are determined by the time period specified in the browse policy. For more information about the browse policy, see "What Are Browse and Retention Policies?" on page 112. If remote access is enabled, users may also be able to recover files from other client computers. For more information on remote access, see “Allowing Remote Access Rights to Other Clients” on page 131.

For specific instructions on using the GUI version of the Backup recovery program (`nwrecover`), refer to the online help.

# Archive and Retrieve Backup Client Applications

Archive is an optional Backup feature that you must purchase and enable separately.

Use the archive feature to save related files or data associated with specific projects at a specific point in time, such as at the end of a project. To start the Backup client application for archive, enter `nwarchive` at the command line.

When Backup archives data, it captures an image of the specified save sets as they exist on the client at that point in time and writes the data to one or more storage volumes, usually removable media (tapes or optical disk). Archived files can be removed from the client computer so the disk space can be used for other purposes.

Use the retrieve feature to get back archived data. To start the Backup client application for retrieval, enter `nwretrieve` at the command line.

For more detailed information on archives, see Chapter 6, “Archive” on page page 137.

For specific instructions on using the GUI versions of the Backup archive and retrieve programs (`nwarchive` and `nwretrieve`), see the online help.

## Permissions Required to Retrieve Archived Data to the Client

The Archive Services attribute in the Clients resource controls whether the archive feature is enabled or disabled for each client. After you have enabled the optional Backup Archive application on the Backup server, archive services are available for all Backup clients that connect to the server. The Archive Services attribute in the Clients resource allows you to restrict the services to selected clients. Once you enable the Archive Services attribute, a user on the client computer can use the nwarchive GUI or nsrarchive program to perform an archive. When you enable or disable the Archive Services attribute for a client, the attribute changes for all Clients resources with the same name.

To restrict the users of the client computer that can request archives, enter user IDs in the Archive User attributes of the Clients resource.

## Permissions Required to Retrieve Archived Data to Another Client

When the Public Archives attribute in the Server resource is enabled, all client computers that have Client resources defined on the Backup server can retrieve the archives of all other client computers defined on that Backup server.

CHAPTER 6

# Archive

This chapter explains how to install and use the optional Backup archive application. The following topics are addressed in this chapter:

- Installation Requirements
- Permissions for Archive and Retrieve
- Archiving Data
- How Backup Performs an Archive
- Retrieving Archived Data to a UNIX Client

## Overview

The Backup archive application (nwarchive) provides archive services to Backup client computers for which you have enabled archiving. The archive process captures files or directories as they exist at a specific time and writes the data to special archive storage volumes. After the archive information is completed, you can delete the original files from the disk (called *grooming*) to conserve space.

Use Backup Archive in addition to scheduled Backup backups to protect your data. Although backups provide short-term insurance against data loss due to hardware failure or user error, archives offer a long-term strategy for data preservation. You can remove archive files from primary disk storage to make space for newer files, and retain them in archive volumes for quick retrieval in the future.

Archived data is never subject to automatic recycling, so Backup cannot accidentally overwrite archived data. Entries for archive volumes are maintained in the media database indefinitely. To restore archived data to local disk, use the Backup retrieve program (nwretrieve).

Because archive data is in a different format than Backup backup save set data, it must be written to different volumes, regardless of whether the archive performed is a PC- or UNIX-style archive.

A PC-style archive provides the option to store client file index entries, which enables you to browse and recover individual files from within the archive save set. You enable this option by selecting the Store Index Entries attribute in the Pools resource you configure for the PC-style archive.

The client file indexes created during a PC-style archive to an archive pool that has the Save Index Entries attribute enabled cannot be written to the same volume as the archived save sets. If you enabled the Store Index Entries attribute for the Pool designated for the PC-style archive, the client file indexes are automatically written to a volume from the Default pool during the next scheduled backup. If you need to direct the client file indexes for the archive to a volume pool other than Default, see “Example: Directing Client Indexes and Bootstrap to a Separate Pool” on page 78 for further information.

A UNIX-style archive does not provide the option to store client file index entries. You cannot retrieve individual files from the archive save set. A unique annotation of 1024 characters or less identifies each archive save set, for retrieval at a later date.

# Installation Requirements

Before you enable the Backup archive program, make sure you have the following:

- An archive enabler certificate for the Backup server.
- Backup Network Edition or Backup Power Edition installed and enabled on the Backup server.
- A device, either stand-alone or in an autochanger or silo, connected to a Backup server or storage node. To clone your archives, you must have two devices available.

## ▼ How to Evaluate Backup Archive

If you already have an authorized copy of Backup software, you must enter a special evaluation enabler to evaluate Backup Archive. The evaluation period is 45 days; Backup Archive stops functioning at the end of the evaluation period.

Follow the instructions listed under “How to Enable Backup Archive” on page 139 to enter the special evaluation enabler code shown in the *Solstice Backup 5.5 Installation Guide and Release Notes*.

**Caution –** You can only use the evaluation enabler code once per network, or it disables all the Backup servers (even for backup) that you enabled with it.

When you purchase Backup Archive, Sun or your Authorized Reseller sends a new enabler code that you can register and authorize for permanent use. To enter your new enabler code, delete the current Archive Support record (which includes the evaluation enabler code) in the Registration resource, then follow the instructions in the following subsection.

## Backup Archive Enabler Entry, Registration, and Authorization

The Backup distribution files (CD-ROM or packages downloaded from the World Wide Web) include the Backup Archive software.

If you are evaluating the Backup software, the archive program is automatically enabled for a 30-day evaluation period. To continue to use the archive program after the evaluation period, you must purchase and enter the enabler code, as described, and register the product.

If you want to evaluate Backup Archive with an enabled Backup server, see the instructions in “How to Evaluate Backup Archive” on page 138.

If you want to use Backup Archive indefinitely, you must follow the instructions explaining how to enable, register, and authorize Backup Archive.

### ▼ How to Enable Backup Archive

1. **Purchase a Backup Archive enabler from Sun or a Sun Authorized Reseller. A certificate with your enabler is sent.**
2. **After you receive the enabler certificate, become root on your Backup server.**
3. **Issue the `nwadmin` command to start the GUI version of the Backup administration program.**
4. **Open the Registration window.**
5. **Click Create.**
6. **Enter the enabler code.**
7. **Click Apply.**

   Register Backup Archive as soon as possible after you enable it. Backup Archive stops functioning if you do not register it within 45 days of entering the enabler.

### ▼ How to Register Backup Archive

1. **In the Backup administration program, display the Server resource in tabular mode.**

2. **Print a copy of the Server resource and mail or fax it to Sun.**

   After you send in your registration information, Sun sends you an authorization code.

### ▼ How to Authorize Backup Archive

1. **In the Backup administration program, display the Registration resource.**

2. **In the Registration resource, select Archive Support from the list.**

3. **In the Auth Code attribute, enter the authorization code you received.**

   See "How to Register and Authorize Your Software" on page 13 for more information about how to enable and register Backup products.

## Permissions for Archive and Retrieve

After you enter the archive enabler code for the Backup server, all clients of that server are enabled for Backup Archive by default. In the Clients resource, you can disable or enable archiving for individual clients in the Archive Services attribute. To archive data that resides on the Backup server, make sure that the Archive Services attribute is enabled in the Clients resource for the server.

**Caution –** When you enable the Archive Services attribute for a client resource, you also enable the Archive Services attribute for all other clients of the same name on that server. For example, if you have a Backup BusinesSuite module and the Backup client software is installed on the same computer that backs up to the same Backup server, both client resources have the same name. The Archive Services attribute is either enabled for both or disabled for both.

You can restrict access to Backup Archive on each client by granting permission to specific users in the Archive Users attribute of the Clients resource. Users listed in the Archive Users attribute can archive any file for which they have read permission, and they can browse the archive save sets (view the annotation information in the media database).

By default, only the owner of an archived save set or the Backup administrator can retrieve that save set. To allow all enabled archive users to retrieve any archived file, enable the Public Archives attribute in the Server resource. Whether you enable or disable the Public Archives attribute, retrieved files retain their original file ownership and access permissions.

You cannot omit specific clients from the retrieval permissions when you enable the Public Archives attribute.

# Archiving Data

On UNIX computers, you can start an archive from either Backup Archive on the client computer or the Backup administration program on the server. In the Backup administration program, you can set up a UNIX client's archive to start immediately or later. On Backup clients of other platforms, you can only initiate an archive from the Backup User client application.

Unlike scheduled backups, scheduled archives only run once. The advantage of a scheduled archive is that you can run the archive at a time when network traffic is low and not tie up computer resources during business hours.

## ▼ How to Request an Archive From a Backup Client

1. **On UNIX client enter `nwarchive` at the command prompt to start Backup Archive. On a PC client, start the Backup User program and select Archive.**
2. **Select the files you want to archive. On a PC client, specify the options to verify, clone, groom, or compress the data.**
3. **Start the archive.**
   **Backup prompts you to enter an annotation. You must enter an annotation before Backup begins the archive. On a UNIX client, in the dialog box where Backup prompts you to enter an annotation, you can also specify options to verify, clone, groom, or compress the data.**

   **The archive starts immediately, if an appropriate archive volume is mounted, and continues until it finishes. If you selected a large amount of data, the archive can take a long time.**

### ▼ How to Schedule an Archive on the Backup Server for a UNIX Client

1. **Start the Backup administration program. Change to a different Backup server, if necessary.**

2. **Create an archive request.**
   - Enter an annotation as part of the archive request.
   - In the Status attribute, select either Start Now or Start Later.
   - In the Save Set attribute, specify the path of the files you want to archive.
   - Specify any options, such as verify, clone, and groom.

   For specific instructions on how to use the Backup administration program, refer to the online help.

3. **Apply your selections.**

   To view information about scheduled archives, such as the status of the archive request and the name of the archive request, display the Archive Request Control resource. If you specified a notification command in the Archive Completion attribute of the Archive Request resource, Backup sends a notification when the archive is finished.

## How Backup Performs an Archive

The implementation of the archive process differs across operating system platforms. Backup uses two types of archives:

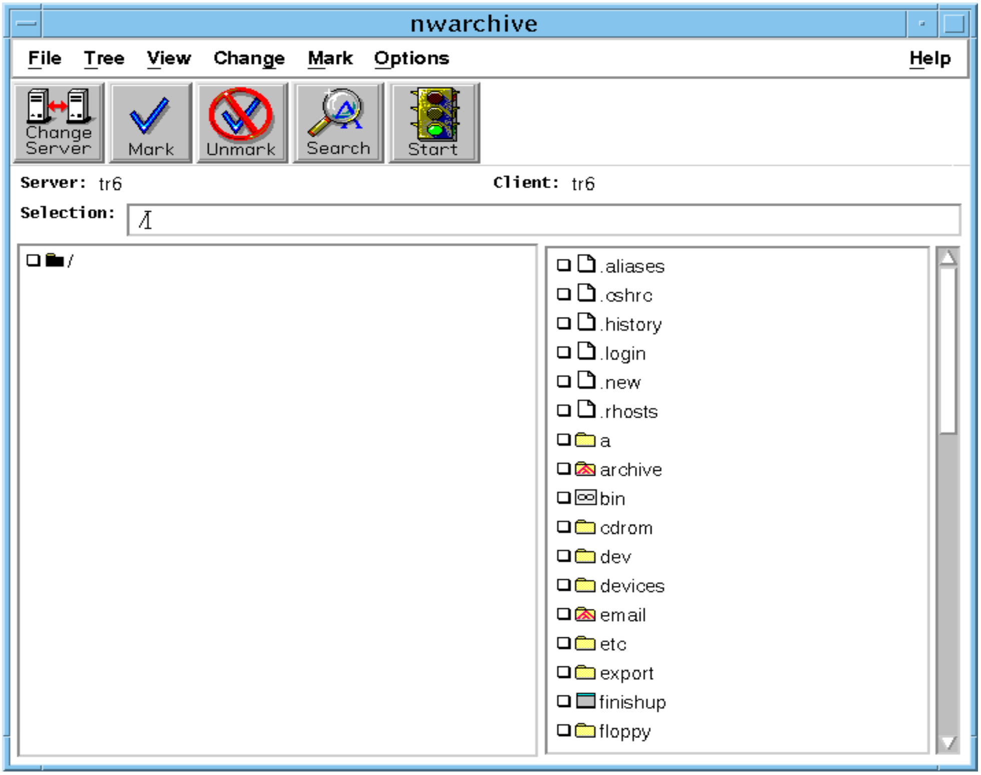
- PC-style archive

  You cannot schedule a PC-style archive from the Backup server; it is initiated by the client's `save` program. This style of archive also gives the user the option to delete the original files after the files are archived. To provide cross-platform compatibility with UNIX servers, the Backup server for UNIX has a preconfigured PC Archive volume pool to receive data generated by archives from non-UNIX clients.

  Refer to the Backup User program online help for instructions on how to make an archive request from a Windows NT or PC client.

- UNIX-style archive

On UNIX, you can use the archive program (nwarchive) for manual archives from the client, or you can schedule archives from the server. FIGURE 6-1 on page 143 shows an example of the main window that is displayed by the nwarchive program. Refer to the online help for details explaining how to use the Backup programs to start an archive.



Server: tr6 Client: tr6

Selection: /

□ /

□ .aliases
□ .cshrc
□ .history
□ .login
□ .new
□ .rhosts
□ a
□ archive
□ bin
□ cdrom
□ dev
□ devices
□ email
□ etc
□ export
□ finishup
□ floppy

**FIGURE 6-1** Main Window for the nwarchive Program

Whether you initiate the archive on the client or the server, the archive is performed by the client's nsrarchive program, which is initiated by the client's nsrexecd daemon. FIGURE 6-2 on page 144 illustrates the UNIX archive implementation.

---

**Caution –** You cannot select the Store Index Entries option in the Pools resource for UNIX-style archives. If you select the option and then apply your changes, you receive an error message. With a UNIX-style archive, you can only retrieve the entire archive save set; you cannot browse or retrieve individual files from within the archive save set.

---

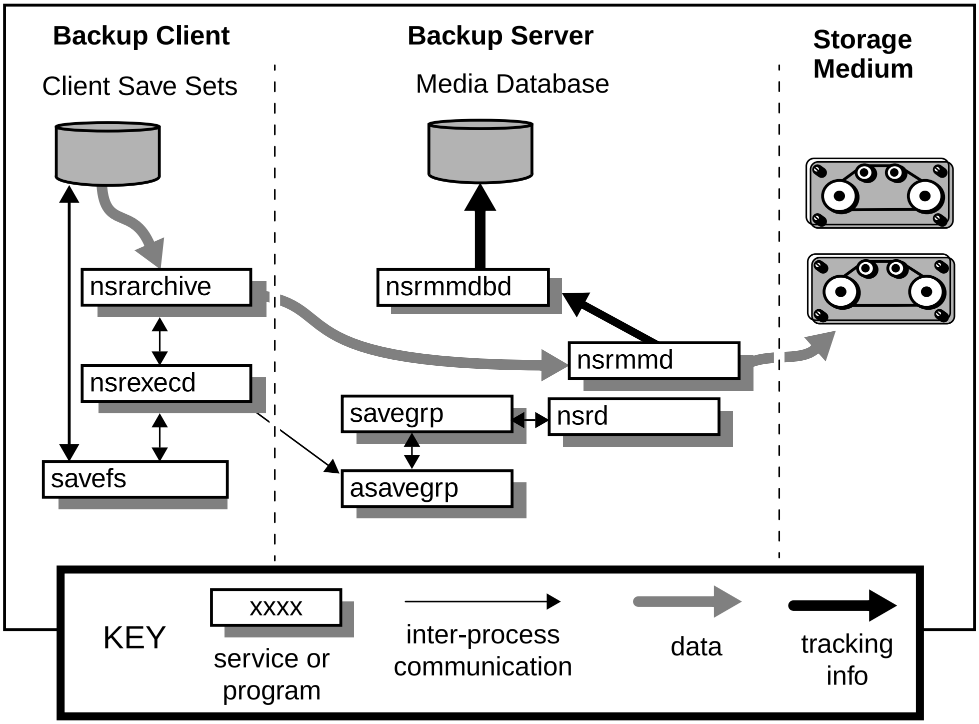


**FIGURE 6-2** UNIX Archive Operation

During the archive operation, the data is written to storage volumes of the archive pool type. The archive volume can be in a device attached to the Backup server or a device attached to a storage node (called a remote device). Information about the archive data, including the annotation that you entered as part of the archive request, is tracked in the Backup server's media database. If you enabled the Store Index Entries attribute in the Pools resource used for a PC-style archive, information about individual files in the archive save set is tracked in the online client file index. The client file index entries that are generated during an archive are backed up during the next scheduled backup, to volumes from the Default pool. Index entries are not made for a UNIX-style archive.

You can select the verification, grooming, and cloning options for an archive operation. If you select verification, Backup checks the integrity of the data on the storage volume against the original data on the client system. If you select grooming, Backup deletes the archived save sets from the source client computer. If you select cloning, a copy of each archive save set is written to a volume from an Archive Clone pool, similar to the process of cloning backup save sets. You can select cloning, verification, and grooming for an archive in either the Archive Options window of the Backup archive program or the Archive Request resource in the Backup administration program for UNIX archives.

# Retrieving Archived Data to a UNIX Client

To copy archived data back to a UNIX client computer, use the Backup retrieve program (`nwretrieve`) or enter `nsrretrieve` at the command line. FIGURE 6-3 on page 146 shows an example of the main window that is displayed by the `nwretrieve` program. To retrieve archived data on a PC client, use the Recover function in the Backup User program. For detailed instructions on how to use the Backup `nwretrieve` and the Backup User GUIs, refer to the online help. For more information about `nsrretrieve`, see "`nsrretrieve`" on page 340 or refer to the `nsrretrieve(1M)` man page.

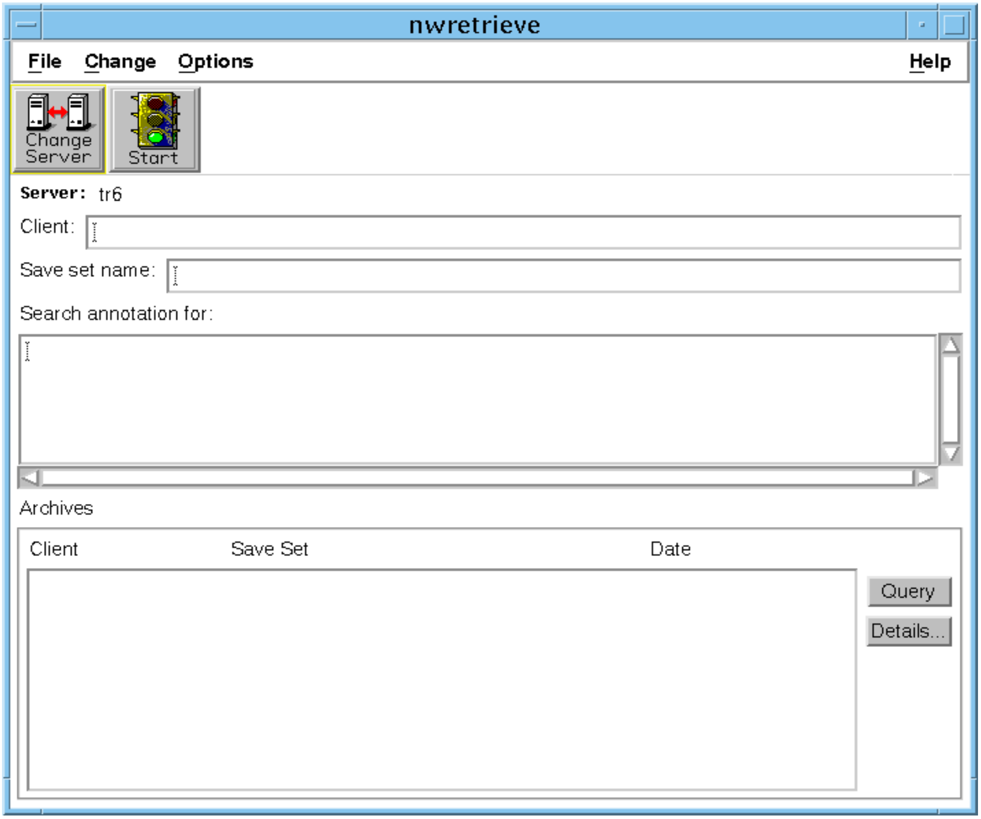


FIGURE 6-3 Main Window for the `nwretrieve` Program

You can retrieve an archive save set if you have administrator or archive user privileges for that save set, or if you enabled the Public Archives attribute in the Server resource. See "Permissions for Archive and Retrieve" on page 140 for more information on Public Archives.

Because archived data are not usually recorded in the online client file index, all the data in an archive save set are retrieved as a single unit. If the Store Index Entries attribute in the Pools resource was enabled at the time a PC-style archive occurs, individual filenames are recorded in the online client file index, and you can use the GUI to browse the files in an archive save set.

When you use the Backup retrieve program, you search the archive save sets based on the client where the save sets originated and on the text in the annotation. Select the archive save set you want to retrieve, then start the retrieval. Before the retrieval begins, Backup prompts you to find out how to handle filename conflicts. Backup also checks whether the archive volumes required to retrieve your data are mounted. If the volumes are not mounted, Backup sends a message according to the configurations in the Notifications resource for tape mount requests.

When the required volumes are mounted, Backup retrieves the save sets you selected. The archived data is still maintained on the archive volume, which remains protected from accidental reuse by Backup. Any entries in the client file index remain unchanged as well.

CHAPTER 7

# Autochangers

This chapter provides information about how to install, configure, and operate Backup autochanger support. See "Device Configuration" on page 71 for additional information pertinent to all backup devices. The following topics are addressed in this chapter:

- Autochanger Installation
- Autochanger Configuration and Management
- Volume Management

## Overview

The Backup software displays the term "jukebox" to refer to an autochanger. The term "autochanger" refers to a variety of robotic libraries, including carousel, library, near-line storage, datawheel, and autoloader.

Autochangers automate the task of loading, mounting, and labeling backup media. Before Backup can back up data to an autochanger, you must:

- Connect and configure the autochanger to the Backup server or storage node computer.
- Install and enable the Backup server or storage node software and device drivers.
- Use the `jb_config` program to configure your autochanger.
- Enable the Backup Autochanger Module.
- Load and label your volumes.

You determine most of the autochanger configuration when you install the Backup device drivers and run the `jb_config` program. After you complete the configuration tasks, you can change the attributes of the autochanger in the Jukeboxes resource.

**Caution –** You cannot add or create autochanger resources using the Backup administration program. You can only modify autochangers previously installed and configured using the `jb_config` program.

If you want to install and use additional autochangers for backups with your Backup server or storage node computer later, you must purchase and enter additional enabler codes to allow Backup to use the additional autochangers.

You use the Backup administration program (`nwadmin` or `nsradmin`) or `jb_config` to modify the configuration of an autochanger or an autochanger device and remove an autochanger from the list available for use by Backup to.

The `nsrd` daemon must be running on the system for which you want to configure an attached autochanger (either the Backup server or a storage node) before you start the `jb_config` program. The installation script provides the option to start the Backup daemons after the installation is completed. You can also start the Backup daemons, as root, from the shell prompt.

# Autochanger Installation

To use an autochanger for Backup storage management, you must first use the `jb_config` program to configure the autochanger and test the device driver software you installed. Follow the instructions in this section to configure and test the device driver software on a Backup server or storage node with a SCSI (Small Computer System Interface) autochanger attached.

**Caution –** For HP-UX, you cannot use the lus drivers provided with Backup. Before you configure an autochanger for use with Backup, you must first install and configure the drivers supplied by Hewlett-Packard. Refer to the documentation CD for instructions on how to configure support for the Autochanger Module with an HP-UX system.

## ▼ How to Configure an Autochanger

To configure an autochanger, follow these steps:

1. **Become root on the Backup server.**
2. **Enter the** `jb_config` **command.**

3. **When the installation script is displayed, enter your response for each query.**

The following example shows the required responses to configure a SCSI autochanger on a Solaris system. The driver software detects and displays the information for all SCSI autochangers attached to the system.

```
# jb_config
1) Install a SmartMedia Jukebox.
2) Install an Autodetected SCSI Jukebox.
3) Install an SJI Jukebox.
4) Install an RLM Jukebox.
5) Install an STL Silo.
What kind of Jukebox are you installing? 2
These are the SCSI Jukeboxes currently attached to your system:
1) scsidev@1.2.0: DLI Libra Series
2) scsidev@0.2.1: Quantum DLT/Digital DLT
Which one do you want to install? 2
Installing a 'Quantum DLT/Digital DLT' jukebox.
Name you would like to assign to the jukebox device? QuantumDLT_1
Pathname of the control port for the jukebox device?
[scsidev@0.2.1] [Return]
Do you want automated device cleaning support enabled? (yes/no) n
Enter pathname of media drive 1 [/dev/nrst8]:? [Return]
This media device has not been configured yet. Please select a
media device type for /dev/nrst8.
a) himt
b) qic
c) 4mm
d) 8mm
e) 8mm 5GB
f) 3480
g) dlt
h) vhs
i) optical
Choice? g
Jukebox has been added successfully
#
```

The following example shows the responses to configure an SJI autochanger.

```
# jb_config
1) Install a SmartMedia Jukebox.
2) Install an Autodetected SCSI Jukebox.
3) Install an SJI Jukebox.
4) Install an RLM Jukebox.
5) Install an STL Silo.
What kind of Jukebox are you installing? 3
#
```

```
Enter the number corresponding to the type of jukebox you are installing:

            1) ADIC-1200c/ADIC-1200d   22) Qualstar

            2) ADIC-VLS                23) Spectralogic

            3) ARC DiamondBack         24) STK 9704/Lago 340

            4) Sun 20Gb 4mm Tape       25) STK 9708/Lago 380
            Loader                     (SCSI) Datawheel

            5) Breece Hill             26) StorageTek 9730

            6) Breece Hill Saguaro     27) StorageTek 9738

            7) Philips Blackjack       28) Dell PowerVault 130T

            8) DLI Libra Series        29) Hewlett-Packard A4853A

            9) Quantum DLT/Digital DLT 30) IBM 3570

            10) Exabyte 10e or 10h     31) IBM 7331/IBM 9427

            11) Exabyte 10i            32) ATL/Odetics SCSI

            12) Exabyte 18D            33) HP-Optical

            13) Exabyte 60             34) Sony TSL-7000

            14) Exabyte 120            35) Digital 4mm DAT TLZ9L

            15) Exabyte 210            36) Digital 4mm DAT
                                       (TLZxx)

            16) Exabyte 220            37) Digital TL800 Series

            17) Exabyte 218            38) Digital TL810 Series
```

```
18) Exabyte 400 Series          39) Digital TL820 Series

19) HP-C1553A/Surestore         40) Digital Optical
12000e

20) HP-C1557A/Surestore         41) Digital TK Series
12000e

21) Metrum (SCSI)               42) Standard SCSI Jukebox
```

```
Choice? 9
Installing a 'Quantum DLT/Digital DLT' jukebox.
Name you would like to assign to the jukebox device? dlt
Pathname of the control port for the jukebox device? scsidev@1.2.0
[Return]
Do you want automated device cleaning support enabled? (yes/no) n
Enter pathname of media drive 1 [/dev/nrst8]:? [Return]
This media device has not been configured yet. Please select a
media device type for /dev/nrst8.
a) himt
b) qic
c) 4mm
d) 8mm
e) 8mm 5GB
f) 3480
g) dlt
h) vhs
i) optical
Choice? c
Jukebox has been added successfully.
```

When you use the `jb_config` program to configure an autochanger, Backup creates a new resource with the name you specified. You can view the new resource in the Jukeboxes resource in the Backup administration program. Refer to the online help or the `nsr_jukebox(5)` man page for details on the attributes of the Jukeboxes resource.

Note that the choices for devices and media types that are displayed might differ from the examples shown, depending on whether you install device supplements or patches to the Backup software after the Backup software is installed.

## ▼ How to Test the Autochanger Connection

To test the autochanger connection, follow these steps:

1. **Become root on the Backup server or storage node.**
2. **Insert two volumes, one each into the first and last slots of the autochanger. Make sure that the drives are empty and that any drive doors are open.**
3. **Enter the `jbexercise` command at the prompt; specify the control port and the device type.**

   The control port for SCSI autochanger models is typically expressed in the format `/dev/scsidev@n.n.n`. You can obtain the exact control port pathname from the response displayed by the `jb_config` command script:

   ```
   These are the SCSI Jukeboxes currently attached to your system:
   1) scsidev@1.2.0: DLI Libra Series
   2) scsidev@0.2.1: Quantum DLT/Digital DLT
   ```

   For example, the following command runs the `jbexercise` program on the Quantum DLT/Digital DLT autochanger detected by the `jb_config` script:

   ```
   # jbexercise -c /dev/scsidev@0.2.1 -m “Quantum DLT/Digital DLT”
   ```

   See Chapter , “Command Line Reference Utilities” on page page 261 or refer to the `jbexercise(1m)` man page for additional information on the command options available for the `jbexercise` command.

## ▼ How to Enable and Register the Autochanger Software Module

After you install, configure, and test the autochanger, enter the enabler code for the Backup Autochanger Software Module according to the instructions on your enabler certificate. Be sure to register and authorize the Autochanger Software Module, or the software disables itself 45 days after you enter the enabler. See “Enabler Code Entry” on page 12 for further information.

If you install additional autochangers later, you must enable and register each additional Autochanger Software Module you purchase as well as configure and test the driver software for the new autochanger. You only need to reinstall the Backup device drivers to add an additional autochanger if you removed the device driver software after the original installation.

## Remote Autochanger Management

Autochangers that are connected to storage node computers require a few additional configuration and management steps.

You can control most operations on remote autochanger devices from the Backup administration program. But for some remote autochanger operations, such as reset, you must use the `nsrjb` or `jb_config` commands on the storage node computer. You can issue the commands as root, either from the local computer or through a remote login session.

After you install the storage node binaries on the storage node computer, define the storage node's devices. The method for defining devices is described in "Remote Device Configuration" on page 75. An overview is provided here.

When you add a remote autochanger device, first add the storage node's hostname to the Administrator attribute in the NSR resource (Server window in the Backup administration program) in the following form:

```
root@storage-node-hostname
```

Then, run the `jb_config` program on the storage node computer (as shown in "Autochanger Installation" on page 154) to define each device in the autochanger. See Appendix B, "Command Line Reference Utilities on page page 261 or refer to the `jb_config(1m)` man page for the syntax and options for this program.

The device names for remote devices begin with a prefix of `rd=` and the name of the storage node computer. For example, `rd=omega:/dev/rmt/1mbn` is a device called `/dev/rmt/1mbn` on a storage node computer called *omega*.

# Hints and Tips for Specific Autochanger Models

This section provides tips to use and configure several supported autochanger models.

## Tips for the Optical Autochanger

- If your optical autochanger does not work with Backup, your operating system might not currently include support for the optical autochanger media drive.

- When you create the Backup device names for an optical autochanger, you should use the raw name of the device.
- If you experience a power outage, the optical autochanger loses track of the state of its backup volumes. Issuing the **nsrjb -H** command might not successfully reset the autochanger. If you see output similar to the following, it means the **nsrjb -H** command failed to reset the autochanger:

```
# nsrjb -H
nsrjb: Autochanger error, Invert operation not supported
```

Use the following procedure to reset the autochanger:

1. Enter the **nsrjb** command without the **-H** option at the system prompt, and make a note of the output. This command displays information about the volumes loaded in the drive(s), the volume label(s), and the slots from which the volumes were loaded into the drive(s).

2. Manually unload the volume from the device. Consult your autochanger hardware documentation for information on how to do this.

3. Manually reinsert the volume into the cartridge access port, flipping them over so that the “A” side faces up.

4. Use the control panel on the autochanger to reload the volume from the cartridge access port into the slot it originally came from.

5. Enter **nsrjb -H** at the system prompt to reset the autochanger.

## Tips for the ADIC 1200c Autochanger

The ADIC 1200c does not allow you to set the SCSI address of the autochanger (robotic arm) and the tape drive separately. You can only set the SCSI address of the tape drive. The SCSI address assigned to the autochanger is always three numbers higher than the tape drive address.

## Tips for the HP C1533A and HP C1553A Autochangers

The following information applies specifically to the use of an HPC1533A or HP C1553A autochanger with Backup.

## Dip Switch Settings

To ensure that the HPC1533A and HP C1553A autochangers function correctly with Backup, refer to *Technical Bulletin 144*, included in the `bulletins.pdf` file. The option switches are located on the underside of the drive.

## Transfer Rates With Solaris

If you experience poor transfer rates with an HP C1553A attached to a computer with the Solaris operating system, refer to *Technical Bulletin 142* for instructions.

# Using a Hewlett-Packard Tape Drive With IRIX

If you want to use the HP C1553A 4 mm tape drive or robotic unit, for example, the Spectra Logic SL-4000 autochanger with your IRIX Backup server or storage node, you must insert the following lines in your `/var/sysgen/master.d/scsi file`, after the entry listed for the Archive Python drive:

```
[ DATTAP, TPDAT, 2, 6, “HP”, C1533A”, 0, 0, [0, 0, 0, 0], MTCAN
_BSF|
MTCAN_BSR|MTCAN_APPEND|MTCAN_SETMK|MTCAN_PART|
MTCAN_PREV|MTCAN_SYNC|MTCAN_SPEOD|MTCAN_CHKRDY|
MTCAN_VAR|MTCAN_SETSZ|MTCAN_SILI|MTCAN_SEEK|
MTCAN_CHTYPEANY,
40, 8*60, 4*60, 5*60, 512, 512*128}
```

After you insert these lines, rebuild the kernel and reboot the computer with the following commands:

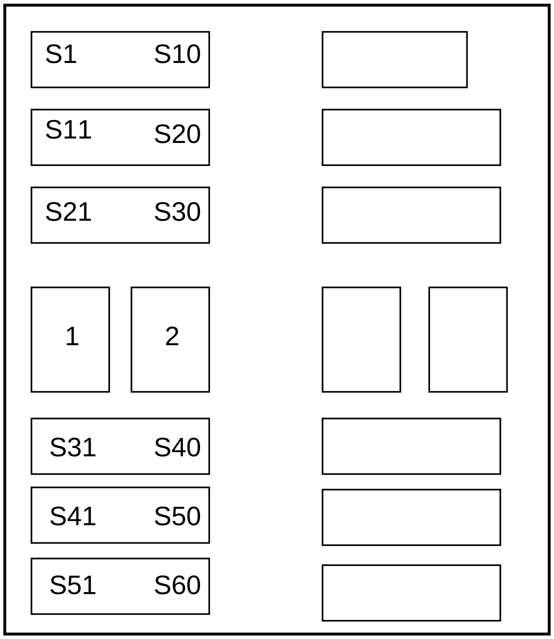
```
autoconfig -f
reboot
```

HP-C1533 drives as configured at the factory will not work with your Silicon Graphics system. A bank of eight switches is located on the bottom of the drive. SGI recommends that you set switches 1, 2, and 5 on and leave the remaining switches turned off.

# Tips and Suggestions for the EXB-60 CHS

The following tips and suggestions apply specifically to the use of an EXB-60 CHS with Backup.

The slot and drive addressing in the EXB-60 CHS is as follows:

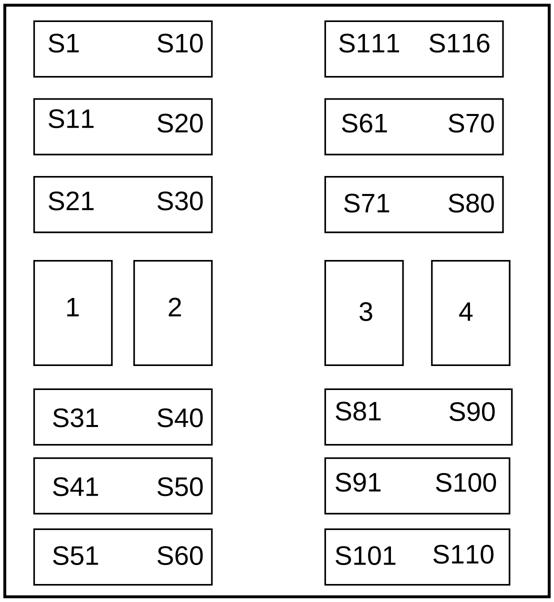


The EXB-60 CHS has the “element status” feature. Keep the autochanger loaded with volumes to optimize the inventory process. If you insert or replace a volume in a slot, you must issue the `nsrjb` command with the initialize element status (`-E`) option to update the autochanger inventory. If the device encounters an empty slot during the inventory, it rechecks several times to make sure the slot is truly empty. This increases the time it takes to inventory the slots.

# Tips and Suggestions for the EXB-120 CHS

The following tips and suggestions apply specifically to the use of an EXB-120 CHS with Backup.

The slot and drive addressing in the EXB-120 CHS is as follows:



If you install less than four media drives in the EXB-120 CHS, you must install the drives in the first physical positions in the autochanger. For example, if two drives are to be installed, they must be placed in the first and second positions. If three drives are to be installed, they must be placed in the first, second, and third positions.

The EXB-120 CHS has the "element status" feature. Keep the autochanger loaded with volumes to optimize the inventory process. If you insert or replace a volume in a slot, you must issue the `nsrjb` command with the initialize element status (`-E`) option to update the autochanger inventory. If Backup encounters an empty slot during the inventory, it rechecks several times to make sure the slot is truly empty. This increases the time it takes to inventory the slots.

# Autochanger Configuration and Management

After you install and test your autochanger on a server or storage node computer, use the instructions in this section to manage the devices and media in your autochangers.

## ▼ How to Add More Autochanger Devices

To add more devices to an autochanger, first define the new devices in the Devices resource so the Backup server or storage node recognizes the added devices. Then, enter the pathnames for the autochanger devices in the Jukeboxes resource so the Backup software recognizes that the new devices belong to the autochanger.

---

**Caution –** After you add a device pathname to the Devices resource, you must also add it to the Devices field in the Jukeboxes resource. Backup associates the device in the autochanger with the autochanger name. Refer to the Backup online help for a description of the Devices resource and how to use it.

---

If an autochanger has more than one device, you must list the device pathnames in the same order as their physical location in the autochanger. If you are unsure of their order, refer to the autochanger hardware manual or become root on the Backup server or storage node computer and enter the `inquire` command at the shell prompt. The `inquire` command returns a list of the SCSI devices attached to your system.

---

**Caution –** The `inquire` command is only supported for AIX, Solaris, and DYNIX/ptx systems. To view a list of the SCSI devices attached to an HP-UX system, enter the `ioscan -f` command. Refer to the documentation CD for instructions on how to configure support for the Autochanger Module with an HP-UX system.

---

# Auto Media Management With Autochanger Devices

The auto media management feature further automates the backup procedure. It frees you from the task of prelabeling volumes before backup. When you enable the auto media management attribute in the Jukeboxes resource, Backup assumes that the volumes loaded in the autochanger can be handled exclusively as Backup media. Backup volumes that appear unlabeled are considered blank and are automatically labeled, mounted, and overwritten with new data. Backup only uses volumes that it considers unlabeled if it cannot locate a writable volume or a volume ready for recycling.

Backup does not recognize the following volume labels and considers these volumes unlabeled and available for use:

- Volumes without a label.
- Volumes labeled with something other than a Backup label.

- Volumes with Backup labels that were written in a density different from the device in which it is currently loaded. For example, if you performed Backup backups on an older 8 mm tape drive, a newer 8 mm tape drive might not be able to read the volumes from the older device because a different density was used to write the data.

---

**Caution –** Be careful about sharing Backup volumes between different backup devices if you have auto media management enabled. You could potentially relabel and overwrite valuable data. If you place or store other volumes you do not want Backup to use in the autochanger, place them in slots that are not included in the available slot range assigned for Backup to use.

---

The auto media management attribute is located in both the Jukeboxes and Devices resources. For autochangers, you only need to enable the Auto Media Management attribute in the Jukeboxes resource. Backup does not allow you to enable Auto Media Management in the Devices resource if the device is located in an autochanger.

If you back up to a stand-alone device and want to use the Auto Media Management feature, see “Standalone Device Configuration” on page 72.

## Cleaning Cartridge Use and Management

Maintaining a backup device in good working order requires periodic cleaning. Backup provides automatic cleaning of devices located in an autochanger. Backup does not support automatic cleaning for stand-alone devices.

The Sun Backup Device Supplement contains a list of currently supported autochangers for which Backup automatically recognizes cleaning cartridges.

Use both the Jukeboxes and Devices resources to make the necessary selections for automatically cleaning your autochanger devices.

The choices specific to the autochanger appear in the Jukeboxes resource, where you enable and disable the automatic cleaning feature. Select the appropriate cleaning slots for the cartridges.

The functions specific to the devices located in the autochanger appear in the Devices resource. You are notified when a device needs cleaning and of the date the device was last cleaned, and can decide how often a device should be cleaned.

Backup only cleans devices before mounting or after unmounting a volume in a device to ensure that there is no interference with other autochanger operations.

The Backup cleaning cartridge support provides notification messages to inform you of cleaning cartridge operations, as shown in Table 7-1.

**TABLE 7-1** Auto Clean Notifications

| Notification Message | Meaning |
|---|---|
| Device cleaning required | The Auto Clean attribute is disabled and the device needs to be cleaned. |
| Device cleaned | The Auto Clean attribute is enabled and the device has been cleaned. |
| Cleaning cartridge required | The Auto Clean attribute is enabled and there are no usable cleaning cartridges available. |
| Cleaning cartridge expired | The cleaning cartridge has been used the specified number of times and needs to be replaced. |

Check the documentation from your autochanger manufacturer for recommendations on the frequency and cleaning methods for your autochanger devices.

## ▼ How to Use a Non-Default Slot for the Cleaning Cartridge

To use a non-default slot for the cleaning cartridge, follow this steps:

1. **Insert the cleaning cartridge in the slot you select.**
2. **In the Jukeboxes resource, find the Default Cleaning attribute and write down the number of uses left on the cleaning cartridge. To view the Default Cleaning attribute in the Backup administration program, display the Jukeboxes window in details mode.**
3. **Specify the slot number you want to use for the cleaning cartridge in the Cleaning Slots attribute, and set the Auto Clean attribute to Yes.**
4. **Change the Available Slots attribute to reflect the range of slots available for data volumes.**

   If the slot for your cleaning cartridge is not the first or last slot in the autochanger, you must specify two ranges of slots in the Available Slots attribute. This is because the inventory of the autochanger must be performed in two steps, once for each range of slots.

For example, if the autochanger contains 11 slots and slot 6 is used for the cleaning slot, specify

```
1-5
7-11
```

on separate lines in the Available Slots attribute, or use the following syntax with **nsradmin**:

```
available slots: 1-5, 7-11;
```

5. **At the command line, enter the following command:**

```
nsrjb -U uses -j autochanger -S slot
```

Replace *uses* with the number of uses left on the “Default Cleaning” field (the number you wrote down in Step 2) and replace *slot* with the slot you now use as the cleaning slot. You can omit the `-j` option if there is only one autochanger.

Every time you replace the cleaning cartridge in the autochanger, you must run the **nsrjb** program to specify the number of uses left.

**Caution –** If your autochanger does not support the element status or barcode labeling option, you must issue the command shown in Step 5 to tell the autochanger to add the cleaning cartridge to its inventory.

# Cartridge Access Port

A Cartridge Access Port (CAP) enables you to deposit and withdraw volumes in an autochanger without opening the door to the autochanger. Each time you open the door of an autochanger to add or remove media, you invalidate the status of the autochanger. You then need to reinventory the contents of the autochanger so that Backup can track the backup media. The inventory process can take a long time to complete.

This CAP feature is useful because you can add (deposit) and remove (withdraw) volumes in an autochanger without having to reinventory the autochanger.

When you use the CAP to add or remove volumes, Backup does not automatically take inventory, read barcode labels, or locate empty slots in the autochanger. Use the autochanger inventory feature and Jukeboxes resource for these tasks.

## ▼ How to Deposit a Volume

To use the CAP to deposit a volume, follow these steps:

1. **Become root on the Backup server or storage node computer.**
2. **Press the button on the front of the autochanger to move the cartridge holder forward and open the CAP.**
3. **Place the volume in the holder.**
4. **Press the button again to move the media into the autochanger and close the CAP.**
5. **Enter the `nsrjb` command at the system prompt. Replace *slot* with the slot number:**

```
# nsrjb -d -S slot
```

If you do not specify a volume name with the `nsrjb` command, Backup marks the slot with a "-*" to indicate that the volume in that slot is unknown. Inventory the slot with the following command:

```
# nsrjb -I -S slot
```

To verify that the volume was deposited in the correct slot, use Backup to mount the volume.

## ▼ How to Withdraw a Volume

To use the CAP to withdraw a volume from a specific slot in the autochanger, follow these steps:

1. **Become root on the Backup server or storage node computer.**
2. **Enter the `nsrjb` command at the system prompt. Replace *slot* with the slot number for the volume and *port* with the value assigned to the control port:**

```
# nsrjb -w -S slot -P port
```

3. **When the volume reaches the CAP, press the button to open the CAP.**
4. **Remove the volume and close the CAP.**

# Volume Management

The topics in this section provide instructions for several common volume management tasks. You can use the Backup administration program (nwadmin), the nsradmin interface, or the nsrjb program to perform volume management tasks.

For specific instructions on how to use the windows in the Backup administration program, refer to the online help. For details on the nsradmin and nsrjb commands, refer to the nsradmin(1m) and nsrjb(1m) man pages. The nsrjb command is also described in Appendix B, "Command Line Reference Utilities" on page page 261.

## ▼ How to Mount a Volume in an Autochanger

To mount a volume in an autochanger, select the autochanger device in the Devices attribute and then mount the device, or enter the following command at the shell prompt:

```
# nsrjb -l -S slot
```

Place an adhesive label on the outside of the autochanger to identify its device pathnames. When you use more than one autochanger, this practice is especially useful to remind you which device pathnames belong to the autochanger.

## ▼ How to Label a Volume in an Autochanger

Select a backup device in an autochanger from the Devices attribute, then select a label template in the Pools resource, and make sure there are volumes in the autochanger. Then start the volume label operation. To label volumes in an autochanger from the command line, enter the following command at the shell prompt:

```
# nsrjb -L -S slot-range
```

You can add another `-S` designation with another slot range to specify more than one slot range on the same `nsrjb` command line. If you want the `nsrjb` program to label a volume on a specific device, add the `-f` option and list the device name. For a complete description of all the options that are available with the `nsrjb` program, refer to the `nsrjb` man page.

Because it takes Backup some time to label the volumes in the autochanger, you might want to perform the volume label process when you do not need to back up or recover files.

Backup starts to label the media in the autochanger with the label displayed in the Starting With attribute. The First Slot and Last Slot attributes determine the range of slots containing volumes that Backup labels.

If you label a specific range of volumes, the name in the Starting With attribute must match the label template. If you label a single volume, you can use any name; it does not have to match the label template. To label a single volume, put the same value in the First Slot and Last Slot attributes.

When a valid Backup label already exists on the media that you are trying to label, Backup displays a confirmation message to keep you from accidentally relabeling the media. When a volume is relabeled, you cannot recover its contents under the previous label. When you select OK to confirm, the volumes in the slots are loaded, labeled, and unloaded.

---

**Caution –** When you label volumes, first unmount all volumes currently in the autochanger to prevent you from accidentally relabeling existing volumes when you reload the autochanger with new volumes.

---

For more information on labeling volumes see “Storage Management Operations (Labeling and Mounting)” on page 86.

## How Backup Uses Barcode Labels With Autochangers

The use of external barcode labels to label media provides two distinct advantages: it significantly speeds up volume inventory and provides improved accuracy for internal volume labels.

With barcode labels, the inventory operation is fast and efficient because you do not have to load the volumes into the device. Instead, the autochanger scans the external barcode labels with an infrared light while the volumes remain in their slots. Performing an inventory with barcode labels greatly reduces the time it takes to locate a volume or determine the contents of a volume.

Barcode labels also provide greater accuracy because the labels are attached to the media prior to being loaded and scanned in the autochanger. After the autochanger scans the barcode label, Backup records and tracks the label in the media database.

Backup only uses barcode labels to inventory volumes. Backup uses the internal volume label (usually created with a label template) to identify the volumes required for backup and recovery. However, Backup displays both the barcode label and the volume label in the pending messages, and the Volumes resource contains both the volume label and its associated barcode label.

You do not have to label existing volumes with barcode labels if they are stored in a vault or offsite for long periods at a time. This is because you do not inventory these volumes often, if ever. However, if you have volumes you use often for recovery or to be overwritten with new data, it is beneficial to label them with barcode labels. When your volumes are labeled with barcodes, you save hours of time when you inventory your volumes.

If you decide to use barcode labels on your existing volumes, you must first apply the barcode labels to the existing volumes. Then, load and mount each volume individually so Backup can match the barcode label to the existing volume label.

You can purchase a variety of barcode labels from a third-party vendor. You can choose numeric or alphanumeric labeling, or a special combination of numbers and characters to meet your labeling needs. You can even order barcode labels that match your current volume labeling scheme.

If you label your volumes with the server name and an extension such as “001,” order a range of labels starting with “server_name.001” and ending with “server_name.100.” Labeling instructions for barcode labels are usually provided with your autochanger hardware documentation. If you have questions about barcode labels, contact the hardware manufacturer.

Using a consistent labeling scheme helps you better organize and track your volumes. It also aids the inventory process if all the volumes, rather than a limited number of them, use barcode labels.

When Backup relabels volumes automatically, it reuses the original volume label name. You can only change the label name if you relabel the volumes manually. Backup scans the barcode label during the labeling process and updates the media database with the new volume name and its associated barcode label.

If the autochanger inventory becomes outdated, either by rebooting your system or by opening the autochanger door, you can update the information about the autochanger’s contents by performing an inventory. The administration program provides a graphical Inventory command; you can also issue the `nsrjb -E -I` command, as root, at the shell prompt. A reset operation also updates the information about the contents of the autochanger. Regardless of which method you use to update the contents, every slot in the autochanger is initialized.

## ▼ How to Label an Autochanger Volume With Barcodes

A volume must have a volume label, but it does not require a barcode label. Use the Jukeboxes resource to associate barcode labels with your volumes.

To label Backup volumes with barcode labels, follow these steps:

1. **Apply the barcode labels to your volumes.**
2. **Place the volumes with the barcode labels in the autochanger.**
3. **Display the Jukeboxes resource.**
4. **Set the Barcode Reader and Match Barcode Labels fields to Yes.**
5. **Label the volumes using either the Backup administration program or** `nsrjb -L`**.**

---

**Caution –** Do not use identical barcode labels for any of your Backup volumes. Using identical labels defeats the purpose of using barcode labels. If you try to label a second volume with an identical barcode label and you enabled Match Barcode Labels in the Jukeboxes resource, Backup displays an error message and does not allow you to label the second volume. To correct the problem, apply a different label and begin the labeling process again.

---

If you choose not to match the volume label to the barcode label, you should create and attach volume labels to the outside of your media and label volumes in the following manner:

1. **Attach all the barcode labels to the media, then load the volumes in the autochanger.**
2. **In the Jukeboxes resource, set the Barcode Reader attribute to Yes, and set the Match Barcode Labels attribute to No.**

   If you set both Barcode Reader and Match Barcode Labels to Yes and you forget to attach a barcode label, you receive an error message that says there is no barcode label for that volume.

3. **Begin the labeling process. Backup uses the next available label from the label template for the volume name. Backup labels the volumes and records both labels in the media database.**
4. **After Backup completes the labeling process, display the Volumes resource to determine the volume label and barcode label for each volume. Create your own volume labels to attach to the volumes.**

Use the information in the Volumes resource to match the correct volume labels to the barcode labels. This is the easiest way to make sure that you attach the correct volume labels to the volumes with barcode labels.

## How the Inventory Process Works

When Backup labels the contents of an autochanger, it registers the location of the volumes in the autochanger slots when it assigns the volume label. As long as you do not change the volumes in the autochanger after labeling them, Backup can access the volumes because each volume label is assigned to a specific slot.

However, if you change the contents of the autochanger without performing the labeling process, or if you move volumes into new slots, you must inform Backup that the autochanger now holds a different set of labeled volumes or that the volumes are in a different order. This is called taking inventory.

When you inventory the volumes in the autochanger, Backup reads the label of each volume and records its slot number. For example, if you have more than one job pack for an autochanger, you must take inventory each time you remove one job pack and load another one into the autochanger, if you do not label the volumes in the new job pack.

Backup provides the capability of reading barcode labels to speed up the inventory process. We strongly recommend that you use barcode labels if you have a large number of volumes or change your autochanger contents often. See “How Backup Uses Barcode Labels With Autochangers” on page 170 for more information.

## ▼ How to Inventory Volumes in an Autochanger

To inventory volumes in an autochanger when you have moved or added volumes you can either start the inventory operation in the Backup administration program or at the command line (enter `nsrjb -Iv`).

After completing an inventory, Backup registers the contents of the autochanger and then proceeds with its network-wide backup and recovery services.

## ▼ How to Determine Which Volumes Are Used for Backup

The Available Slots attribute enables you to control which volumes Backup uses for backing up data. Backup uses all of the volumes in the autochanger for recoveries. However, you can designate a range of available volumes in the autochanger to control which volumes are selected for Backup backups.

For example, you might designate slots one through five for your Backup backups in an autochanger that contains ten slots. The entries can be a range of slot numbers or a single slot number.

With two-sided media, the number of available slots is always one-half the number of labels (or sides of the optical disks). For example, if you have 32 optical disks, labeled atlas.001.a to atlas.032.b, you have 64 labels (and 64 sides). However, the number of available slots is 32.

For a silo library, the value listed in the Available Slots attribute does not indicate a numbered slot inside the silo. Slot numbers in silos indicate the volume's position in the list of silo volumes. The Backup software uses the value in Available Slots to determine which volumes in the silo are reserved for Backup to use, since at most sites, a silo is shared among several applications.

---

**Caution –** Make sure you place volumes in all the available slots of the autochanger so Backup can proceed uninterrupted with an automatic backup.

---

## ▼ How to Disable the Element Status Feature

Your autochanger may support the element status feature. This feature, along with the bar code label feature, speeds up the inventory process by eliminating the need to load and read volume labels. Backup uses the element status feature to determine whether a slot contains a volume. If the volume has a bar code label, Backup reads it without loading the volume.

If you inventory an autochanger containing volumes without bar code labels, Backup must load the volume to read the internal label. In this case, the element status feature is not beneficial, and you should disable the element status feature.

Check the documentation from your autochanger manufacturer for information on features of your autochanger.

To disable the element status option for your autochanger:

1. **Display the Jukeboxes resource in Details view (**`nwadmin`**) or show the hidden options (**`nsradmin`**).**
2. **In the Jukebox Options attribute, enter:**

```
!element_status
```

## ▼ How to Check Autochanger Notifications

Backup uses e-mail to send notices about Backup events. The software uses the three Tape Mount Request notifications to inform you that the autochanger needs attention. The Tape Mount Request 1 notification is undefined so you can create your own notification message.

The following situations require attention:

- The volumes in the autochanger are 90% full.
- The autochanger needs more volumes to continue.
- The autochanger has a mechanical problem.
- The autochanger device needs cleaning.
- The cleaning cartridge in the autochanger needs replacing.

The Notifications resource contains tape mount and device cleaning request notifications pertaining to autochanger operations.

See "Preconfigured Notifications" on page 64 for more information.

If the volume is loaded in the autochanger, Backup can automatically mount the correct volume so that the recovery proceeds. If Backup requires volumes for a recovery that are not loaded in the autochanger, you receive a notice in the Pending window of the Backup administration program.

After you correct an autochanger problem, you might need to mount a volume before continuing to back up or recover files. Check the Pending window in the Backup administration program for messages.

# Suggestions for Operating Autochangers

This section provides additional suggestions to help you use your autochanger and backup media effectively and reliably.

## Recycling Versus Adding More Backup Volumes

Backup can save files on volumes marked “appen” (appendable) in the Volumes resource. If the volumes inside the autochanger are marked “full,” they cannot receive additional backups.

You can do one of the following with volumes marked full:

- If you need to keep the volumes for long-term storage, remove the full volumes and replace them with new media.
- If you do not need the data on the full volumes, you can manually change the mode to “recyc” (recylable) in the Volumes resource. Backup overwrites the data with new backups, but maintains the existing labels. This is the only instance in which you do not need to relabel a volume to make it eligible to be overwritten with new data.

The mode of a volume is automatically changed to recyclable when all the save sets on the volume have passed the time period specified by its retention policy.

There are advantages to both recycling media and adding more media to a pool. By recycling, you reuse the same volumes and do not add new volumes to the pool. However, the media can wear out over time and have a higher failure rate.

On the other hand, if your site requires that you maintain the backups in storage for a certain period of time, then you might have to add more media to the volume pool instead of recycling it. For example, an autochanger might need new volumes every three months if your company policy is to maintain the backups for a year. In this case, you have to keep adding new media to the pool until you can recycle the volumes that contain expired or old backups.

## Using Pools With an Autochanger

If you plan to have full and nonfull backups, we recommend that you estimate the number of volumes needed for your full backups and assign them to the Full pool. This ensures that your full backups are in a consecutive range of slots in the autochanger, which makes it easy for you to remove all the volumes at the same time.

## Calibrating the Devices

Check the autochanger manufacturer’s documentation for information about the frequency and method for calibrating the loading mechanism for the autochanger device.

## Moving Media

Always use Backup to move the media inside an autochanger. If you physically move the media, the autochanger inventory becomes outdated. If this occurs inadvertently, follow these steps:

1. **Reset the autochanger with the following command**:

```
# nsrjb -H
```

2. **Inventory the autochanger contents with the following command**:

```
# nsrjb -I -E
```

Refer to Appendix B, “Command Line Reference Utilities” on page page 261 and the `nsrjb` man page for details on the `nsrjb` program.

CHAPTER 8

# Hierarchical Storage Management

This chapter includes instructions to configure the Solstice Backup HSM or Solstice Backup HSM XDSM software on clients of the Backup server. It also includes information regarding automatic migration and manual migration over NFS. See "Automatic Migration" on page 188 for details on configuring a Backup client as a migration client. See "Manual Migration of XDSM HSM Files over NFS" on page 202 for details on performing migration operations on a computer that is not a Backup client, but is serviced by an NFS server that has been configured as a Backup migration client.

This chapter also includes information for upgrading a Solaris HSM client from release 5.1 or earlier to XDSM HSM. See "How to Convert Legacy Symbolic Links to New XDSM Stubs" on page 186 for this upgrade information. The following topics are addressed in this chapter:

- HSM Software Installation on a DIGITAL UNIX Client
- XDSM HSM Software Installation on a Solaris Client
- Enabling and Registering the HSM Module
- Initial XDSM HSM Configuration
- Automatic Migration
- Configuring a Migration Client
- Testing File Migration
- File Migration Management

## Overview

Hierarchical Storage Management (HSM) allows you to keep newer data available for fast access without compromising the availability of older or less frequently accessed data. The Backup software supports two different implementations:

- Backup HSM, which supports DIGITAL UNIX clients through the use of symbolic file links.

- Backup X/Open Data Storage Manager (XDSM) Hierarchical Storage Management, which provides support for Solaris clients over network file systems (NFS). The XDSM HSM software uses the industry standard Data Management Application Programming Interface (now known as XDSM, but formerly known as DMAPI) to ensure that file migration is entirely transparent to applications and users both locally and over network file systems (NFS).

## Supported Features

The HSM software module supports the following features for migrated files that retain their unmigrated file size and attributes. However, migrated files use much less physical disk space.

The XDSM HSM software module supports these additional features for migrated files that:

- Remain regular UNIX files, as do the stub files left on the client file system.
- Can be recalled transparently over NFS. No special NFS software is required.
- Have a file *fingerprint* size that is configurable for each file system.
- Have file privileges and attributes that can be accessed or modified without recall.

## How HSM Complements Backup Operations

HSM is a complementary solution to backup, archiving, and staging. HSM allows system administrators to manage network resources more effectively, often resulting in lower cost for hardware storage. All Backup features store data on media; each one, however, has a specific purpose.

The following table compares the goals of backup, HSM/XDSM HSM, and archiving to demonstrate how these features work together to provide a complete storage management solution.

**TABLE 8-1** Comparison of Backup, HSM/XDSM HSM, Staging, and Archive Operations

| | Backup | HSM/XDSM HSM | Save Set Staging | Archive |
|---|---|---|---|---|
| Goal | Protects data from accidental loss or damage | Conserves network storage resources | Moves data from one storage medium to another | Conserves online storage space |
| Files stored | Entire file system | Infrequently accessed files | Any backed up, migrated, or archived file | Rarely accessed files |
| Frequency | Regularly | Policy-based | Policy-based | Usually at project's end |
| Method | Automatically | Automatically or manually | Automatically or manually | Manually |
| Original File | Left in place | Stub remains; can recall file | Moved to new storage medium | Usually deleted |
| Method used to return file to client | Restored by administrator if data online is corrupted or accidentally deleted | Recalled automatically and transparently whenever user tries to access file | Restored by administrator if data online is corrupted or accidentally deleted | Retrieved by administrat or if needed by users |

# HSM Software Installation on a DIGITAL UNIX Client

When you installed the client software subset on your DIGITAL UNIX client, the HSM software was also installed. There is no need to install additional software. To provide HSM services to a DIGITAL UNIX client, purchase and enter an HSM enabler code on the Backup server (see "Enabler Code Entry" on page 12). Once the enabler code is entered, configure a Migration resource for each DIGITAL UNIX client for which you want to provide HSM services, through one of two interface options:

- The Migration window from the Clients menu available through the `nwadmin` GUI.

- The NSR Migration choice from the Create menu available through the `nsradmin` interface.

See “How to Configure a Migration Client Resource” on page 194 for more information on how to configure the Migration resource.

# XDSM HSM Software Installation on a Solaris Client

The XDSM HSM software is a three-part process. To complete the process successfully, you must:

- Install the Backup client software on the Backup client computer.
- Install the SUNWshsm  package on the Backup client computer. If you want to migrate and recall files over NFS, this computer must also be the NFS server.
- Install the SUNWshsmn package on the migration client or any Solaris NFS client from which you want to be able to explicitly migrate and recall files remotely.

The following figure illustrates the location of each software package in a typical network configuration. In this scenario, host Oak is the Backup server and has the XDSM HSM module enabled. Host Elm is an NFS server that also functions as a Backup backup client and as a migration client. Additionally, Host Elm has the XDSM HSM packages installed on it. Depending on your environment, the Backup server might be the same physical computer as the NFS server.

Hosts Birch and Pine are NFS clients of the NFS server, Elm. Both Birch and Pine have the migration command line utilities (the SUNWshsmn package) installed, allowing them to issue remote migration and recall operations on Elm.

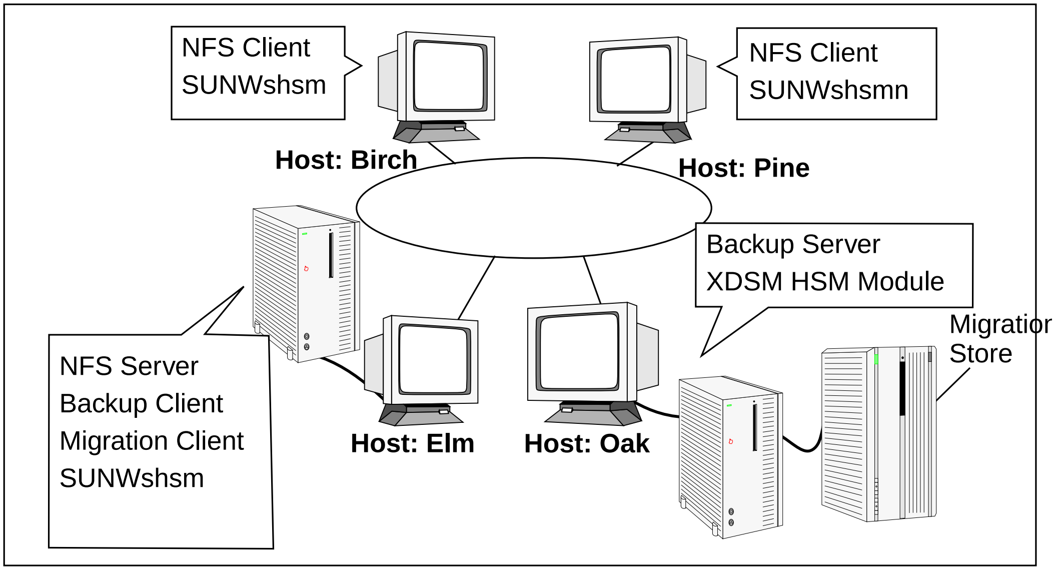


**FIGURE 8-1** Configuration Scenario for XDSM HSM Software

---

**Caution –** This version of HSM is not compatible with the previous version of HSM based on symbolic links. If you have migrated files created with a version of HSM prior to the 5.2.1 XDSM HSM release, you must use the `sym2xdm` program to convert the symbolic links to XDSM stubs. See "How to Convert Legacy Symbolic Links to New XDSM Stubs" on page 186 for more information.

---

## Files Installed During XDSM HSM Installation

The XDSM HSM installation process uses two software packages to install the XDSM HSM software on a Solaris client:

- SUNWshsm
- SUNWshsmn

The SUNWshsm package is installed on the migration client. The SUNWshsmn package is installed on one or more Solaris NFS clients or the migration clients to enable the Solaris NFS clients to issue explicit migration and recall commands.

- Two kernel modules and a kernel-level pseudo-device driver are installed on the migration client computer when you install the SUNWshsm package. Together, these files (listed as follows) implement an extended version of the industry standard, XDSM:

  - stackfs, a stacking mechanism that sits on top of your existing file system and allows access to the XDSM HSM software
  - xdm, a StackFS module that implements an extended version of XDSM
  - dm_plc, a pseudo-device driver that enables communication between the XDSM library and the kernel-level code
- The following are the Backup HSM binaries with the new XDSM versions, installed in the /usr/sbin directory:
  - nsrexecd
  - nsrhsmck
  - nsrib
  - nsriba
  - nsrmig
  - nsrpmig
  - dmib
  - dmls
  - dmrecall
  - dmclear
  - sym2sdm
- hsmnfsd, a new executable that enables NFS clients to explicitly execute migration operations on the migration client in /usr/sbin.
- A shared library file that communicates between the XDSM HSM save, recover, nwbackup, and nwrecover programs in the /usr/lib/nsr directory:
  - hsmip_save.so.1
  - hsmip_recover.so.1

The following files can be installed with the SUNWshsmn package on one or more Solaris NFS clients. The command line utilities installed with this package allow Solaris NFS clients to explicitly migrate, recall, and generate statistical reports about a remotely mounted filesystem that contains migrated files. These files are:

- Three new command line utilities in the /bin directory on a Solaris NFS client for remote migration, recall, and reporting operations:
  - migrate, the remote migration utility
  - recall, the remote recall utility
  - migls, the remote reporting utility that shows the migration state of a set of files
- Man pages for the new command line utilities in the /usr/share/man/man1 directory.
- The conversion utility:
  - sym2xdm, conversion utility for symbolic links to XDSM stubs
  - fndlink, lists all symbolic links for the previous HSM version

# Enabling and Registering the HSM Module

When you purchase the Backup HSM or Backup XDSM HSM module, an Enabler Certificate with an enabler code for your copy of the software is mailed separately from the product package. See “Enabler Code Entry” on page 12 for instructions on how to enable, register, and authorize the software for permanent use.

# Initial XDSM HSM Configuration

Before you begin to use XDSM HSM to migrate your files, you must first configure the Backup server to recognize the client computer as a migration client.

## ▼ How to Mount the StackFS Layer

For the Backup server to recognize the client computer as a migration client, you must remount each local UFS or VxFS file system that you wish to enable for migration onto the same mount point using the StackFS stacking mechanism that you just installed:

1. **Become root on the migration client computer.**

2. **Ensure that the UFS or VxFS filesystem is idle and clear of all other users. (See the** `fuser` **man pages for information to perform this task.)**

3. **Remount each local UFS or VxFS filesystem that you want to enable for migration onto the same mount point that uses the StackFS stacking mechanism.**

   For example, if your UFS or VxFS filesystem is mounted on `/home`, enter the command:

```
# mount -F stackfs -o xdm /home /home
```

4. **You can ensure that this remount step occurs when you reboot the computer by using your preferred text editor to add the following line after the existing UFS or VxFS mount entries in the** `/etc/vfstab` **file:**

```
/home - /home stackfs - yes xdm
```

**Caution –** Automating the remount step to occur when you reboot ensures that your users can always access a file system containing migrated files controlled by XDSM HSM.

5. **See "Automatic Migration" on page 188 for details on configuring and using XDSM HSM.**

## ▼ How to Unmount the StackFS Layer

If you no longer want a filesystem to be under XDSM HSM control, you can unmount the filesystem from the StackFS layer. For example, if the filesystem is mounted on `/home`, use the command:

```
# umount /home
```

This command leaves the physical disk partition mounted as a standard UFS or VxFS filesystem on `/home`. To completely unmount the UFS or VxFS filesystem, you need to run the `umount` command again.

Then remove the following line after the existing UFS or VxFS mount entries in the `/etc/vfstab` file:

```
/home - /home stackfs - yes xdm
```

## ▼ How to Convert Legacy Symbolic Links to New XDSM Stubs

The `sym2xdm` program is a conversion tool that converts symbolic links created by 5.1 Backup HSM or earlier to the new XDSM file stubs.

If you are running 5.1 Backup HSM or earlier and you have migrated files, the new XDSM will not recognize the migrated files’ symbolic links. For the new XDSM to recognize the migrated files, the symbolic links must be converted to XDSM stubs.

Use `sym2xdm` to do this conversion, as follows:

1. **Become root.**
2. **Stop the Backup daemons with the command:**

```
# nsr_shutdown -a
```

3. **Mount all file systems containing symbolic links from 5.1 Backup HSM or earlier with the command:**

```
# mount -F stackfs -o xdm /home /home
```

(See “How to Mount the StackFS Layer” on page 185.)

4. **List the existing HSM symbolic link with the command:**

```
# fndlink /home
```

This lists all symbolic links used for HSM on the console. If you want to save these to a text file, type:

```
fndlink /home > filename
```

5. **Restart the Backup daemons with the command:**

```
# /etc/init.d/networker start
```

6. **Convert the symbolic links to XDSM file stubs with the command:**

```
# /usr/sbin/sym2xdm -s server-name /home
```

A list of the converted files is displayed on your screen.

7. **If you want to verify that all symbolic links have been converted, repeat Step 4 for all directories that contain HSM files.**

# Automatic Migration

The HSM software enables you to effectively manage a network's storage resources by allowing you to keep newer data available for fast access without compromising the availability of older or less frequently accessed data. The software module must be enabled on a Backup server and the Backup client computer must be configured as a migration client.

Information on the following topics appears in this section as noted below:

- "How Automatic Migration Works" on page 188 describes migration functionality.
- "Configuring a Migration Client" on page 193 discusses the configuration tasks you should perform to set up a migration client.
- "File Migration Management" on page 197 provides information on additional administrative tasks and issues you should consider when you use the XDSM HSM software with Backup.

## How Automatic Migration Works

HSM automatically moves data between your local disk and another storage medium based on a set of policies specified by an administrator. File *migration* is the process of moving files from a client to the *migration store*, which is a remote migration storage device; *recall* is the process of moving files from the remote storage device back to the original location on the client. The purpose of HSM is to manage a network's storage resources more effectively by keeping newer data available for fast access without compromising the availability of older or less frequently accessed data. Except for a relatively longer access time for migrated files, the entire migration and recall process is transparent to you and your applications.

HSM moves files between the migration client and the migration store and is managed by the migration server. The *migration client* is any system on the network containing data to be migrated. The *migration server* is a system on the network providing migration services. The migration store is attached to the migration server and can consist of disks, tapes, or optical storage media.

This data management strategy relies on the administrator's definition of a *high water mark*, which defines the threshold condition that determines when automatic migration begins, and a *low water mark*, which defines the threshold condition that determines when automatic migration stops. Migration continues until all eligible files are migrated or until the low water mark is reached.

## File Migration

File migration is a “sweeping” operation determined by the criteria you define. Backup generates lists of files that are candidates for migration according to the assigned criteria. Access time is the most frequently used parameter to determine these candidates. You can enable or disable migration services for each migration client. Certain files are always excluded from migration. These files include system files, shared libraries, and all executables and data files used by Backup.

File migration can be either automatic or manual, depending on the requirements of your system. All you have to do is define your criteria and assign the appropriate criteria to each migration client. Backup automatically migrates each file that meets those criteria on the client. Backup automatically recalls a migrated file when a user or application reads or writes past the *fingerprint* of the stub file size.

When a file is migrated, the original file on the client computer is replaced with a stub file that points to the location of the migrated file on storage media. The stub file contains information about the original file and serves two purposes:

- As a place holder for the migrated file, making it appear as though the file is still resident on the local disk
- As a pointer to the new location, allowing the HSM software to find the migrated file and recall it to the local disk

After a file migrates and leaves a stub file, the user can perform the same actions on the stub file as on any other file in the file system. The stub file can be moved, renamed, or have any other action applied to it that does not require read or write access, for example, changing attributes such as owner, group, mode, and timestamp.

### Migration Policies

You can set the values in the Migration resource that HSM uses to determine what the client file system capacity should be for migration to start and stop. For each migration client, you determine the following:

- High water mark – specifies the percentage of disk space filled. When this value is reached, migration starts automatically.
- Low water mark – specifies the percentage of disk space filled after migration. When this value is reached, migration stops.

When the client file system reaches the specified high water mark, the Backup HSM/XDSM HSM application automatically migrates the files that meet the defined criteria.

In addition to the high and low water marks, you set the criteria that files must meet to become candidates for migration. If you set more than one criterion, files must meet *all* the specified criteria to become candidates for migration. For example, you can set a policy that specifies that when the client file system exceeds 70% full, files in the `/home` directory that have not been accessed in over 60 days *and* are at least 2 KB or larger in size are automatically migrated. You can set the following migration criteria:

- Last access time – specifies the length of time since a file was last accessed.
- Minimum file size – specifies the minimum file size to consider for migration. Files smaller than this entry do not provide enough available disk space after being replaced with a stub to warrant migrating them.
- File owner – specifies the name of the owner of the file you want considered for migration. If you want all owners allowed, leave this text box blank. If you want all owners allowed except for *owner_name*, enter *-owner_name* in the field.
- File group – specifies the name of the group with access to the files to be migrated. If you want all groups allowed except for *group_name*, enter *-group_name* in this field.
- Preserve – specifies the files you do not want migrated. These entries may contain UNIX shell wildcard characters.

## How Files Are Migrated

After you have specified migration policies for your migration client in the Migration resource, Backup migrates files in the following way:

1. When the Backup server conducts a regularly scheduled backup, it checks each client in the backup group for files that are candidates for migration. During a scheduled backup, the premigration command, `nsrpmig`, searches the migration client filesystem for files that meet the migration criteria.

   Premigration is a resource-intensive activity. The group containing migration clients should start its scheduled backup at a time of low system use.

2. Files that meet the migration criteria are premigrated. During premigration, the file is copied to a Backup storage location (a migration volume), but the original file remains on the client computer.

3. When the client file system reaches the high water mark, the `nsrexecd` daemon starts the migration command, `nsrmig`, and migration occurs automatically. The `nsrmig` command checks the premigrated files to ensure that they still meet the migration criteria. If the premigrated files are still candidates for migration, the `nsrmig` command does the following:

   - Renames the original file on the client file system with a temporary name.

- Creates a stub on the client file system to point to the migrated file on the migration media. See "How to Configure the XDSM HSM File Fingerprint Size" on page 192 for details on how to change the size of the stub file for XDSM HSM clients.
- Deletes the original file from the client filesystem.

4. Migration continues until the low water mark is reached. If not enough files meet the migration criteria, the migration process might not meet the low water mark.

5. A migration report is emailed to the administrator.

FIGURE 8-2 on page 191 illustrates the interaction between the `nsrpmig`, `nsrmig`, and the Backup daemons, as well as the data movement from the client file system to the migration store.

The Backup server makes entries for migrated files in the client file index. These entries, however, are not visible to a user through the `nwrecover` GUI. The Backup server uses these entries to track the link between the migrated file and the stub in the client filesystem as well as for recall purposes. Because a migrated file must be available for a user to recall, the index entries for migrated data are exempt from the automatic data recycling policies set for a Backup client. See "Migration Monitoring" on page 209 for details on how Backup handles files that have been deleted from the client filesystem.

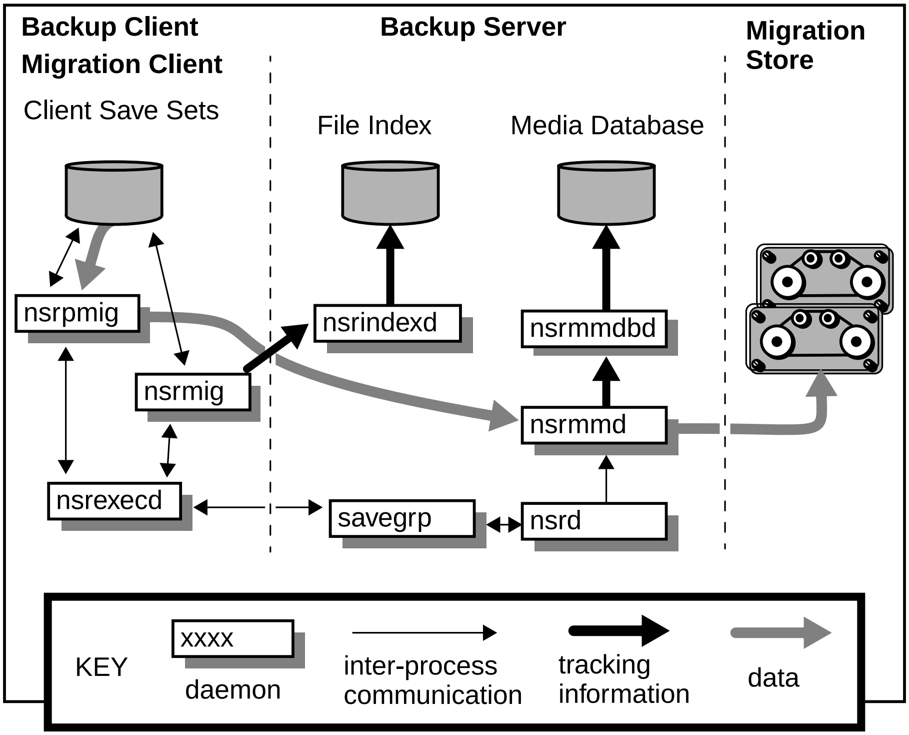


**FIGURE 8-2** Process Interaction and Data Movement During Automatic Migration

## ▼ How to Configure the XDSM HSM File Fingerprint Size

You can specify a file fingerprint size which sets the size of the stub file that remains on the client file system after migration. You can perform read and write operations on a file within the fingerprint area *without* causing the file to be recalled. The file fingerprint size can only be set on a per-filesystem basis and is set in kilobytes. The default file fingerprint size is 32 Kbyte. The minimum value you can set for the file fingerprint size is 1 Kbyte.

To change the value you must include a mount option in the `/etc/vfstab` file. For example, to change the fingerprint size to 64 Kbyte, add the following line after all the existing UFS or VxFS filesystem mount entries in the `/etc/vfstab` file (the following example assumes that the UFS or VxFS file system is mounted on `/home`.):

```
/home - /home stackfs - yes xdm,fp=64
```

The next time you reboot your computer, the 64 Kbytes fingerprint size is used for all future migrated files.

If you want the new fingerprint size to take effect before you reboot your computer, follow these steps:

1. **Edit the `/etc/vfstab` file to include the new file fingerprint value.**

```
/home - /home stackfs - yes xdm,fp=fingerprint-value
```

This example assumes that the UFS or VxFS file system is mounted on `/home`.

2. **Become root on the migration client computer.**
3. **Ensure that the UFS or VxFS filesystem is idle and clear of all other users. You can use the `fuser` command to perform this task.**
4. **Unmount and then remount the `/home` directory using the following commands:**

```
# umount /home
# mount /home
```

Remounting the `/home` directory picks up the new file fingerprint value in the `/etc/vfstab` file. The new fingerprint size is applied to all future file migrations.

## Files That Are Not Migrated

Certain files are always excluded from migration. These files include system files, shared libraries, and all executables and data files used by Backup. The following files are excluded from migration:

- All files in the `/`, `/usr`, `/opt`, and `/var` filesystems
- All files that end with `.so`
- All files (executables and data files) used by Backup
- Files that are larger than 2 GB

Additionally, you can choose certain files or groups of files to exclude from migration. For example, you can exclude files owned by root from automatic migration.

# File Recall

When a user or application accesses a migrated file, either locally or over NFS, to read or write beyond the file fingerprint size, Backup automatically recalls the original file to the location of the stub file. The recall operation must be completed before the read or write can proceed. Therefore, the user might notice a delay in reading and writing until the file is completely recalled to its original position. Other than a slower access time, however, the entire recall process is transparent to the user. Access time depends on the availability of the migration media, device speed, and network speed. See “How to Configure the XDSM HSM File Fingerprint Size” on page 192 for details on changing the file fingerprint value.

If the local hard disk has insufficient free space to recall the file, an error message is generated, Backup sends the appropriate notification message, and a log entry is made in `/var/adm/messages`. Backup provides preconfigured migration notifications that you can customize to suit your environment. See “Notifications Resource” on page 24 for details on using notifications.

# Configuring a Migration Client

After you enable HSM or XDSM HSM on a Backup server, you can configure all the clients managed by that Backup server as migration clients. To ensure that your files are migrated correctly, you should perform the following configuration tasks:

1. Create a group for migration clients.
2. Configure a migration pool resource for your migration data.
3. Configure a migration client resource for each migration client.

## Creating a Group for Migration Clients

You should create a group for your migration clients. The start time for this group determines when scheduled premigration activities take place. When Backup backs up a group containing migration clients and save sets, premigration occurs for qualifying files. Premigration happens automatically and is not controlled by high and low water marks. For automatic premigration to occur, Backup requires that files be premigrated as part of a group.

Since premigration is a resource-intensive process, you should designate a start time for the group when other system demands are low. See Chapter 2, “Getting Started,” on page 11 for details about how to configure Backup groups.

## ▼ How to Configure a Migration Pool Resource

Migrated files are written to a migration-type pool. A migration-type pool differs from both a backup-type pool and an archive-type pool. Because each of the pool types writes data in a different format, you cannot mix backup data and migration or archive data within the same pool. When the HSM or XDSM HSM software is enabled on the Backup server, two additional pool resources are available: Migration and Migration Clone. Use these two pool resources and their corresponding label templates for your migration data. Depending on your needs, you can create several customized migration pool resources for your migration clients. From the `nwadmin` GUI:

1. **Select Pools from the Media menu to open the Pools resource.**
2. **Select the Migration pool or create a new pool resource and designate its Type attribute as Migration.**
3. **Select the migration group you created for your migration clients.**
4. **Apply your settings.**

   During the migration group's scheduled backup, data that qualifies for premigration is written to this migration pool. If you have auto media management enabled or are using an autochanger, Backup mounts a labeled volume from this pool automatically.

## ▼ How to Configure a Migration Client Resource

Use the Migration Client resource for individual migration clients, for each file system on a migration client or for a combination of the two.

---

**Caution –** *Before* you can set up a Migration Client resource you must have a client or client/save set combination resource configured for each client or file system you want to receive migration service. See Chapter 5, "Client Operations" on page page 101 for details on configuring Backup client resources.

---

1. **Ensure that the computer you want to configure as a migration client has been configured as a backup client of the Backup server with the HSM or XDSM HSM software enabled.**
2. **Ensure that** `nsrexecd` **is running on the client.**
3. **Select Migration Setup from the Clients menu to open the Migration window.**
4. **Complete the fields in the Migration window to establish your migration policies. See "Migration Policies" on page 189 for details about the criteria you can designate for your migration clients.**

   Clients and save sets meeting these criteria are available for premigration. Premigration copies the file to the storage location, leaving the original on the client's local disk. When Backup backs up a group containing migration clients, premigration occurs. This happens automatically and is not controlled by high and low water marks. For automatic premigration to occur, Backup requires that files be premigrated as part of a group. When the high water mark is reached or the client file system is full, migration occurs; the premigrated file is deleted from the client file system, leaving a stub that contains information about the migrated file.

# Testing File Migration

To test the HSM software on a DIGITAL UNIX computer, you should test the file migration locally. To ensure that the XDSM HSM software was installed correctly, you should test the file migration both locally and remotely.

## ▼ How to Test Local File Migration

To test local file migration, use the Backup administration program GUI to specify your file migration criteria and then migrate your files manually from the command line. See Chapter 2, “Getting Started” on page 11 and Chapter 4, “Device and Media Management” on page 71 for details on creating groups and pools and on labeling volumes. See “Automatic Migration” on page 188 for a detailed discussion on configuring and using XDSM HSM from the Backup Administrator program GUI.

---

**Caution –** The migration client must be configured as a backup client of the Backup server *before* you can perform the following steps.

---

Make sure the Backup daemons are running. Then, using the Backup Administrator program GUI:

1. **Create a group for your migration files. Be sure to indicate a start time and to enable the Autostart feature.**

   See “Backup Group Configuration” on page 34 for more information.

2. **Create a volume pool resource for migration and put the group you just created in the Migration pool.**

   See “How to Configure a Migration Pool Resource” on page 194 for more information.

3. **In the Migration Setup window from the Client menu, select the group you just created and specify the file migration criteria you want Backup to apply.**

4. **Load and mount a labeled storage volume from the Migration pool in a storage device.**

   See “Storage Management Operations (Labeling and Mounting)” on page 86 for more information.

   If you installed the XDSM HSM command line utilities on a Solaris migration client, you can use the procedure described in “How to Test Remote File Migration With XDSM HSM” on page 197. If you did not install the command line utilities on the Solaris migration client, or if you are testing local file migration for a DIGITAL UNIX HSM client, use the following procedure to test local file migration:

1. **Premigrate a file with the command**:

```
# nsrpmig -s server-name -b pool-name -g group-name absolute-path
```

   The `-b` and `-g` options are not required. If you do not specify these options, the Migration resource defaults are used. If you do not specify a path, Backup uses the current directory.

2. **Migrate a file with the command**:

```
# nsrmig -l 0 -s server-name path
```

   Using the `-l 0` option ensures that the Backup server migrates the specified file even if the file system is below the *low water mark* (the threshold condition that determines when automatic migration stops). If you do not specify a path, Backup uses the current directory.

3. **Verify that a file has migrated by using the** `du -s` **command to compare the file’s premigrated block size with its postmigrated block size. Migrated files have a smaller block size.**

## ▼ How to Test Remote File Migration With XDSM HSM

If you installed the migration command line utilities on an NFS client, you can ensure that they were correctly installed by running the `migrate` command.

1. **Log on to the NFS client computer as root.**

2. **Mount the file system on the NFS server that you want to migrate.**

3. **Make sure an appropriately labeled storage volume is loaded in the storage device.**

4. **Execute the command:**

```
# migrate -s server-name path
```

5. **To view the results of the migration, execute the command:**

```
# migls path
```

See “Syntax for XDSM HSM Migration Command Line Utilities” on page 203 for details on the syntax and options for the command line utilities.

# File Migration Management

This section contains information about additional administrative considerations for your migration clients and migrated files. It includes the following topics:

- Backup and Recovery Considerations With XDSM HSM
- Recovering an Accidentally Deleted Stub
- Cloning Migration Media
- Performing a Super-Full Backup
- Media Management for Migrated Files
- Manually Migrating Files From the Local Migration Client
- Manual Migration of XDSM HSM Files over NFS
- Syntax for XDSM HSM Migration Command Line Utilities
- Migration Monitoring

## Backup and Recovery Considerations With XDSM HSM

After a file has been migrated, when Backup backs up the client file system from this point forward, it only backs up the stub file that is left on the client computer. A Backup backup of a stub file does not recall the migrated file. When you recover a file system that contains a stub file for migrated data, Backup only recovers the stub file to the local disk. The migrated data is not recalled.

To recall a file to the client's local disk, access the file on the client computer beyond its fingerprint or use the `recall` command. Backup automatically recalls the file from the migration store. If the media containing your migrated file is not currently mounted, Backup notifies the administrator. See "Using the XDSM HSM `recall` Command" on page 206 for details on issuing the `recall` command from an NFS client.

# Recovering an Accidentally Deleted Stub

If you accidentally delete an HSM or XDSM HSM stub file on a migration client, you can restore the stub file from backup media within 60 days of the deletion. Recovering a stub file does not initiate a recall. If a stub file has not been recovered after 60 days, Backup removes the entry of the migrated file from the client index and no longer tracks the data.

## How XDSM HSM Handles Deleted Files

The XDSM HSM software includes an automatic index entry clean-up program, `nsrhsmck`, that runs every day. The `nsrhsmck` program performs two operations related to removing unneeded migrated client index entries on the Backup server:

- It scans the time stamp in the Backup client file index for each premigrated and migrated file. For each entry it finds, the `nsrhsmck` program resets the time stamp in the client index to the current date to ensure that the entry does not expire. The `nsrhsmck` program performs this operation by default every morning at 2:00 a.m.
- It scans the client index for expired file entries. If a file's time stamp is older than 60 days, the index entry is deleted from the client index and the file can no longer be recalled. The `nsrhsmck` program performs this operation by default every Sunday at 12:30 a.m.

After you delete a stub file from an XDSM HSM-managed file system, the `nsrhsmck` program does not detect the file during its nightly check and does not update the file's time stamp in the client index. After 60 days, if the `nsrhsmck` program still does not detect the file, its entry is deleted from the client index and the file can no longer be recalled. If, however, you recover the stub file before 60 days have passed, the `nsrhsmck` program detects the recovered stub file during its next nightly check and updates the file's time stamp. As long as a stub file has an entry in the client index, the migrated file can be recalled.

Because the `nsrhsmck` program runs automatically every day, it is unlikely that you will need to run the program manually. If, however, you want to run the `nsrhsmck` program from the command line, the basic syntax is:

```
# nsrhsmck -cfnv -s server-name path
```

You must specify a path on the command line when you run `nsrhsmck`. Only files and index entries that fall under the specified path are examined for consistency.

## Cloning Migration Media

To ensure that you can recover all your data, you should regularly clone your migration media. Because Backup only backs up the stub file on the client computer, *not* the migrated data itself, the clone might contain the only extra copy of a file that exists. You can specify cloning to occur automatically after the migration process is completed by selecting the Migration Clone pool in the group resource you created for your migration clients. Clones of migration data must be written to volumes from a pool of type “migration clone.” See “Using the Migration and Migration Clone Pools” on page 200 for more information.

## Performing a Super-Full Backup

To provide additional backup protection for migrated files, you should regularly perform super-full backups of your migration clients. A super-full backup clones both the most recent full backup of a save set and all the migration save sets, so it contains the stub on the client *and* the data in the migration store. To perform a super-full backup, become root on the Backup server, then enter the following command from the shell prompt:

```
# nsrclone -c client-name -N save-set-name
```

## Media Management for Migrated Files

Migrated data is managed by the Backup server and is subject to all of the usual storage management features, such as pools, cloning, and auto media verification. However, because migrated files must be available for recall by the user, migrated data is exempt from the automatic data recycling policies that the Backup server applies to backup data. This means that Backup tracks the location of the migrated files in the client index and media database as long as the stub file remains on the client computer. Backups of the migrated stub files, however, are subject to the Backup server’s data recycling policies. See “Migration Monitoring” on page 209 for more information.

**Caution –** As long as the stub file remains in the file systems, the migrated files must be available for quick recall. Consequently, stand-alone tape drives for migration are not acceptable. Sun recommends using an autochanger or silo for your migration media.

## Using the Migration and Migration Clone Pools

Migration volumes are the media that hold migrated data. You can either use the preconfigured Migration pool to store migrated data or create your own customized migration pool to use as the migration store. You can also automatically clone the volume to which migrated data is sent. Because migration data is written in a different format than regular backup data, migrated data can be written only to storage volumes associated with a pool of type “migration.” Clones of migration volumes can be written only to storage volumes from a pool of the type “migration clone.” Backup provides preconfigured pools called Migration and Migration Clone for your migration data.

## Client Indexes and the Bootstrap Save Set for Migration Data

Migration data is in a different format than regular Backup save set data; therefore, it must be written to different volumes. Because of these differences, the client indexes and bootstrap save set created during a premigration or migration operation are not written to the same volume as the migrated save sets. By default, they are written to a volume from the Default pool. If you need to direct the client indexes and bootstrap to a volume pool other than Default, follow the instructions below.

You can use regular expression matching to direct the client indexes and bootstrap to a different pool than the one to which you send the backup data.

In the following example, the client file indexes are in `/nsr/index`. To send the Backup server’s bootstrap and all the client file indexes from this file system to the same pool, create a pool (in the Pools resource) with the following attributes:

```
name: Index;
pool type: Backup;
save sets: bootstrap, /nsr/index/.*;
levels: ;
```

When the migration group's scheduled backup runs, the migration client save sets are written to a volume labeled for the appropriate migration pools, while the Backup server's bootstrap and `/nsr/index` save sets are written to a separate volume labeled for the Index pool.

# Manually Migrating Files From the Local Migration Client

If you installed the XDSM HSM command line utilities on the XDSM HSM migration client, see "Using the XDSM HSM `migrate` Command" on page 204 for details on migrating files from the command line. If you have not installed the command line utilities on the migration client, you can migrate files manually by using the `nsrpmig` and `nsrmig` commands.

Use manual migration when the local file system is full or nearly full, for example, after you receive a Migration Attention notification. Migrating large files provides the most benefit, because it frees the most local disk space. See "Syntax for XDSM HSM Migration Command Line Utilities" on page 203 for more information about manually migrating files from an NFS client.

To premigrate a file manually, enter the following command:

```
# nsrpmig -s server-name -b pool -g group path
```

- The `-b` and `-g` options are not required. If you do not specify these options, the Migration resource defaults are used.
- If you do not specify a path, the current directory is used.

After you have premigrated a file, you can migrate it manually by entering the following command:

```
# nsrmig -s server-name path
```

If you do not specify a path, the current directory is used. Migration continues until the file system capacity reaches the low water mark specified in the Migration resource.

Refer to the `nsrpmig` and `nsrmig` man pages for more details about these two commands.

# Manual Migration of XDSM HSM Files over NFS

If your NFS server has been configured as a Backup XDSM HSM migration client, you can use the migration command line utilities from an NFS client explicitly to migrate and recall files on the NFS server and to generate reports on migration information.

**Caution –** An NFS client does *not* have to have the command line utilities installed on it to recall a file on an exported filesystem. Recall happens automatically when a user or application—either locally or over NFS—accesses a migrated file to read or write beyond the file fingerprint size. See "File Recall" on page 193 for details on how automatic recall works.

The XDSM HSM software includes three migration command line utilities that perform the following functions over NFS:

- `migrate` – allows an NFS client to migrate files on an NFS-mounted filesystem.
- `recall` – allows an NFS client to recall files on an NFS-mounted filesystem.
- `migls` – allows an NFS client to view the migration statistics of a set of migrated files.

These command line utilities must be installed on each Solaris NFS client to which you want to give explicit migration control and awareness of migration. The NFS client does not have to be a migration client or a Backup client. The NFS server, however, must have the XDSM HSM software installed on it and also be configured as a migration client of the Backup server. See the following section for information on how to use the migration command line utilities. Refer to the *Solstice Backup 5.5 Installation Guide and Release Notes, Solaris Version,* for instructions on installing the command line utilities on an NFS client.

# Syntax for XDSM HSM Migration Command Line Utilities

This section includes the syntax and option flags that you can use with the three migration command line utilities.

**TABLE 8-2** Migration Command Utilities

| Function | Command |
|---|---|
| Migrate a file or directory over NFS | **`migrate`** [**`-BEiLnpqvx`**] [**`-s`** *server*] [**`-N`** *name*] [**`-e`** *expiration*] [**`-f`** *dirfile*] [**`-b`** *pool*] [**`-g`** *group*] [**`-l`** *level*] [**`-t`** *date*] [**`-m`** *masquerade*] [**`-W`** *width*] [**`-C`** *clone_pool*] [*path...*] |

**TABLE 8-2** Migration Command Utilities

| Function | Command |
|---|---|
| Recall a file or directory over NFS | **`recall`** [**`-vR`**] [*path...*] |
| Display statistics about migrated files over NFS | **`migls`** [**`-R`**] [*path...*] |

You can use the flags listed in Table 8-3 with the migration command line utilities.

**TABLE 8-3** Command Line Utility Flags

| Flag | Function |
|---|---|
| **`-E`** | Estimates the amount of data that will be generated by the save, then performs the actual save. Note that the estimate is generated from the inode information, and thus the data is actually read only once. |
| **`-i`** | Ignores any `.nsrhsm` directive files as they are encountered in the subdirectory structures being saved. |
| **`-L`** | Saves will be performed from the local Backup client, even when files are from a network file server. To recover these files, run `recover` with the **`-c`** *client* arguments, where *client* is the name of the Backup client that did the save. |
| **`-LL`** | In addition to treating the backup as a local backup, `-LL` causes an extra line to be printed at the end of the completion output of the form `complete savetime=`*number*, where “number” is the savetime of the save set created by this backup. This option is meant to be used by the `savegrp` command in performing automatic cloning. |
| **`-m`** *masquerade* | Specifies the tag to precede the summary line. This option is used by `savegrp` and `savefs` to aid in `savegrp` summary notifications. |
| **`-n`** | No save. Estimates the amount of data that will be generated by the save, but does not perform the actual save. |
| **`-v`** | Verbose mode. During a recall operation, it causes each pathname to printed to the standard output just before being recalled. |
| **`-q`** | Quiet mode. |
| **`-s`** *server* | Specifies which computer to use as the Backup server. The default is the current computer if it is running the Backup server software; otherwise the computer with the logical name *nsrhost* in the host table is used. |
| **`-N`** *name* | The symbolic name of this save set. By default, the most common prefix of the *path* arguments is used as the save set name. |

**TABLE 8-3** Command Line Utility Flags *(Continued)*

| Flag | Function |
|---|---|
| **-e** *expiration* | Sets the date (in getdate format) when this save set will expire. By default, no explicit save set expiration is used. |
| **-f** *dirfile* | The file from which prototype default directives are read. A *dirfile* of - causes the default directives to be read from standard input. |
| **-b** *pool* | Specifies a particular destination pool for the migration or recall operation. |
| **-g** *group* | This option is used by savegrp and savefs to denote the group of the save and is used by the Backup server to select the specific media pool. |
| **-l** *percent* | Specifies a goal percentage for nsrmig. Migration will stop when the goal percentage is reached. If the goal percentage has already been reached when nsrmig is run, then nsrmig will do nothing and exit. If the -l option is not specified, the goal percentage is read from the appropriate migration client. |
| **-t** *savetime* | Migrates files that were premigrated at *savetime*. |
| **-W** *width* | The width used when formatting summary information output. |
| **-x** | Cross mount points. |
| **-B** | Forces a save of all connecting directory information from root ("/") down to the point of invocation. |
| **-R** | Shows files or recall files in the specified directories recursively. |
| **-C** | Generates a clone of the migrated save set to a specified clone pool. |

## Using the XDSM HSM migrate Command

The migrate command can be issued from a Solaris NFS client to migrate files on an NFS-mounted filesystem. The migrate command migrates a set of specified files and/or directories on the NFS server to the migration store.

When the nsrexecd daemon is started on the migration client computer, it spawns the hsmnfsd daemon. When you execute the migrate command, migrate communicates with the hsmnfsd daemon. The hsmnfsd daemon then runs the nsrpmig and nsrmig commands to premigrate and migrate the specified files. The hsmnfsd daemon initiates one nsrpmig and one nsrmig command for each file or directory to be migrated. Because the hsmnfsd daemon initiates both the nsrpmig and nsrmig commands, the files specified by the migrate command are premigrated, if necessary, and then immediately migrated.

If you installed the command line utilities on the NFS server (migration client), you can use the `migrate` command locally to migrate files.

FIGURE 8-3 on page 206 illustrates the interprocess communication as well as the flow of data when the `migrate` command is executed.

If you do not specify the `-s` *server* option with the name of the Backup server when you run the `migrate` command, the migration client will search the network for the correct Backup server to use. Explicitly using the **`-s`** *server* option helps avoid a potential delay in the migration process. If you do not specify the **`-s`** *server* option, the migration client first checks for the name `nsrhost` in its `/etc/hosts` file. If it doesn't find an entry for `nsrhost`, then it searches the network for the correct Backup server. Adding the alias `nsrhost` to the host file on the migration client can make the migration process run more quickly.

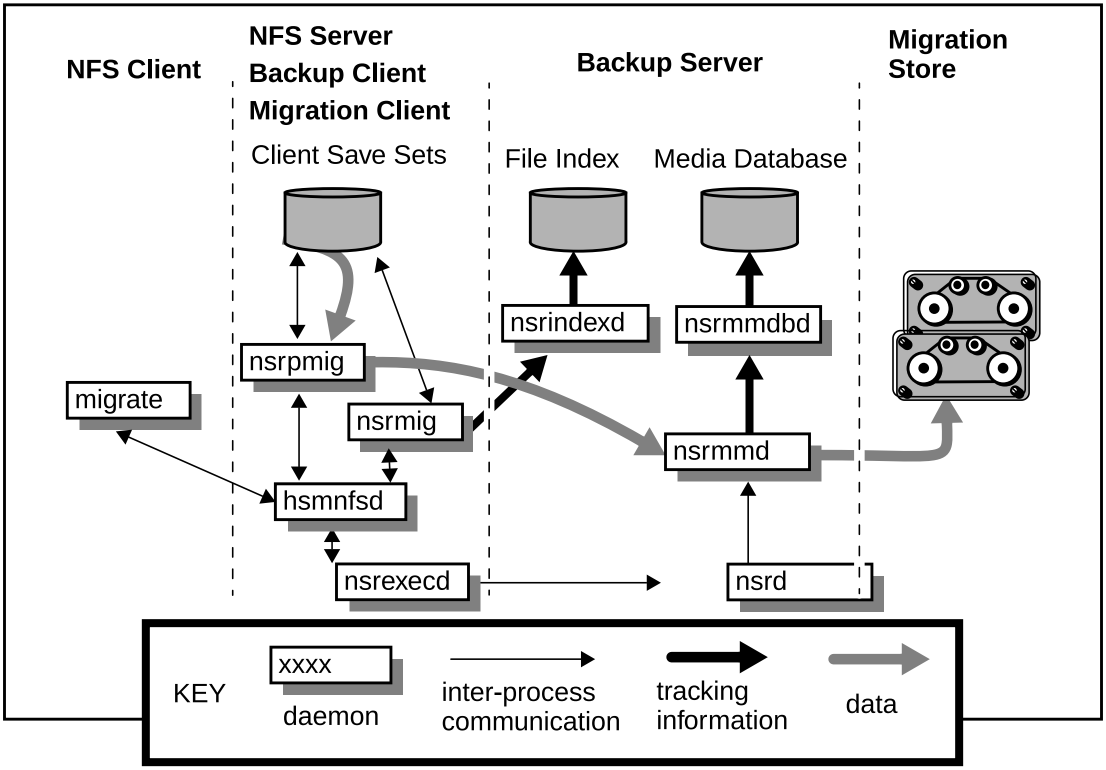


**FIGURE 8-3** Interprocess Communication and Data Movement During a Manual Migration Operation Over NFS

## Using the XDSM HSM `recall` Command

You can recall a file over NFS simply by accessing the file beyond its fingerprint size. Backup automatically recalls the file to its original location. If you want to recall a file without accessing it, you can use the `recall` command from the command line. The `recall` command can be issued from a Solaris NFS client to recall files on an NFS-mounted filesystem. The `recall` command recalls a set of specified files or directories on the NFS server from the migration store. When you execute the `recall` command, `recall` communicates with the `hsmnsfd` daemon on the migration client computer and the file is recalled from migration storage media.

FIGURE 8-4 on page 207 illustrates the interprocess communication as well as the flow of data when the `recall` command is executed.

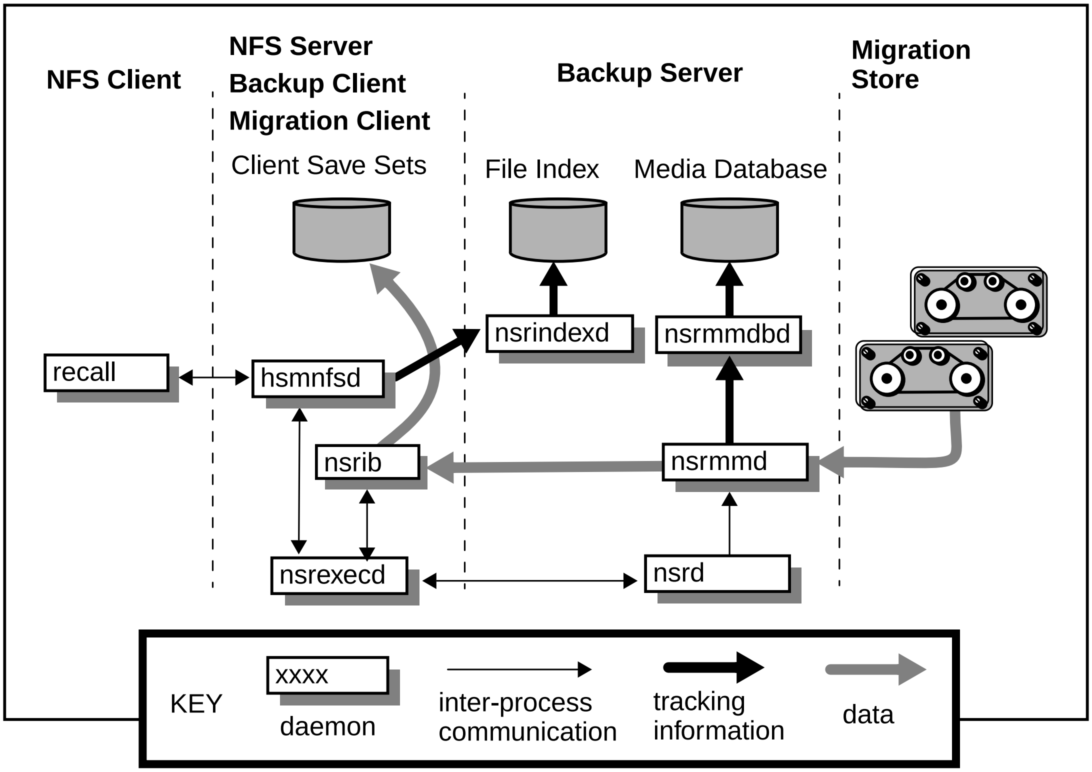


**FIGURE 8-4** Interprocess Communication and Data Movement During a Manual Recall Operation Over NFS

If you do not specify any pathnames or arguments, the `recall` command uses the current directory by default. Unless the `-R` option is included, only files immediately within the specified directories are recalled.

If you installed the command line utilities on the NFS server (migration client), you can use the `recall` command locally to recall files.

Additionally, when a file is recalled, its file access time is updated to the current time. Updating the file access time ensures that a file that has just been recalled will *not* be considered for automatic migration.

## Using the XDSM HSM `migls` Command

The `migls` command enables you to view the migration state of a set of files. The `migls` command displays a list of files and all the files within the given directories. Unless you specify the `-R` option, the `migls` command only shows the contents of the specified direction and does not display the contents of any subdirectories. The pathname for each file in the directory is printed on a separate line and is preceded by one of the following characters:

- `m` – indicates the file has been migrated
- `r` – indicates the file is resident on the client filesystem and has not been migrated
- `x` – indicates a directory or device file

If you do not specify any pathnames or arguments, the `migls` command uses the current directory by default.

If you installed the command line utilities on the NFS server (migration client), you can use the `migls` command locally to view the migration state of your files.

### Example of a Shell Script That Uses the `migls` Command

You can use the `migls` command to create shell scripts with different styles of output. For example, you could write the following script, which generates output similar to the `ls -l` command, except that the first character of the mode field is an `m` if the file has been migrated.

```
#!/bin/sh
    migls $* | while read state path
    do
    case $state in
    m)
        # file is migrated
        ls -l $path | sed 's/-/m/'
        ;;
    *)
        # file is not migrated (or is not a regular file)
        ls -ld $path
        ;;
    esac
    done
```

The output of this script would look similar to the following:

| | | | | | | |
|---|---|---|---|---|---|---|
| drwx------ | 2 root | root | 8192 | Oct 15 | 09:57 | /home/dir |
| -rw-rw-r-- | 1 root | root | 20 | Dec 22 | 14:33 | /home/res |
| mrw-rw-r-- | 1 lr | unixdev | 108894 | Feb 9 | 00:11 | /home/migd |

Depending on the type of information you need, you could create a shell script that would only show migrated files or, similarly, nonmigrated files. For instance, a list of nonmigrated files could be piped into `cpio` to copy only nonmigrated files without causing any migrated files to be recalled.

## Migration Monitoring

The Migration Control resource in the Backup administration program displays a list of clients configured for migration services and statistics for all migration activities that occurred within the last seven days.

You can produce reports on migration activities using command line instructions. Refer to the man pages for details on using these commands:

- Use the `nsrinfo` command to list files in a save set.

- Use the `mminfo` command or the cloning browser to determine which save sets were migrated in the previous twenty-four hours.
- Use the `nsrmig -n` command to produce a report of files eligible for migration without actually migrating them.
- Use the `nsrpmig -n` command to produce a report of files eligible for premigration without actually premigrating them.

To customize the Migration Completion notification, modify the resource configured for the notification. By default, a migration completion notice is sent by email to root any time a migration event occurs, such as the high water mark being reached. See “Event Notification” on page 64 for more information on event notifications.

CHAPTER 9

# SmartMedia

This chapter provides information about how to install, configure, and operate Backup autochanger support. See "Device Configuration" on page 71 for additional information pertinent to all backup devices. The following topics are addressed in this chapter:

- Changes to Backup for SmartMedia
- How Backup and SmartMedia Interact
- Overview of Configuration Tasks

## Overview

Sun SmartMedia™ is a storage management application that manages media resources within a distributed environment. SmartMedia allows your applications to share storage management devices (libraries and drives). For example, SmartMedia allows multiple Backup servers and storage nodes to share common libraries.

The SmartMedia software consists of three main components:

- The SmartMedia server software
- Drive control programs (*DCP*) that control the drives within an autochanger
- Library control programs (*LCP*) that control robotic libraries

The Backup server and SmartMedia server software can exist on the same system (shown in FIGURE 9-1 on page 212) or separately, with SmartMedia running on a remote computer (shown in FIGURE 9-2 on page 213).

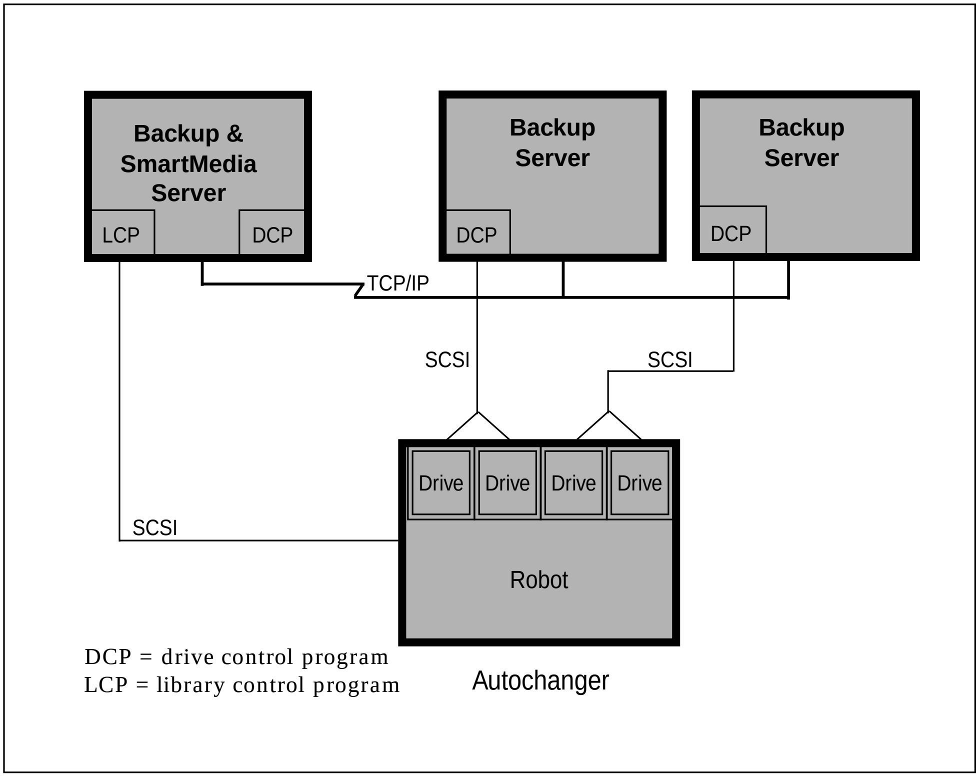


**FIGURE 9-1** Shared SmartMedia and Backup Server

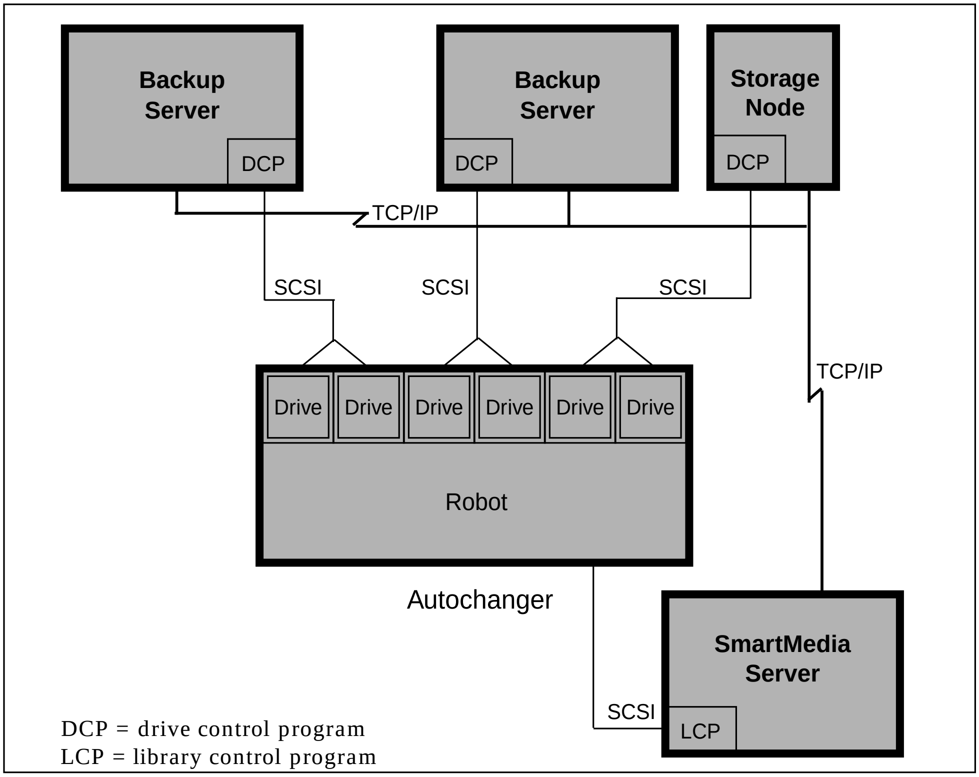


**FIGURE 9-2** Remote SmartMedia Server

## ▼ Purchasing SmartMedia

To use SmartMedia, you need to purchase an autochanger license based on the number of slots your Backup servers and storage nodes will support. Contact SunSoft or your Authorized SunSoft Reseller for more information.

## SmartMedia Documentation

The information in this section is limited to the information required to set up SmartMedia with Backup. In addition to this information, the documentation set for SmartMedia includes the following manuals:

- *Sun SmartMedia Release Supplement*
- *SunSoft SmartMedia Installation Guide*
- *SunSoft SmartMedia Administrator's Guide*

The documentation CD-ROM that accompanies the Backup software package includes the documentation for SmartMedia as well as Backup. In addition, you can access the documentation from `http://www.sun.com`.

### Installation Guide

The *SmartMedia Installation Guide* provides the following information:

- Hardware and software requirements
- Where and how to install the SmartMedia server, LCP, and DCP software
- How to remove the SmartMedia software

### Administrator’s Guide

The *SmartMedia Administrator’s Guide* includes the following information:

- How to set up and configure your SmartMedia server, LCPs, and DCPs
- How to start, stop, and test SmartMedia
- How to share libraries and drives across your network

### Release Supplement

The *SmartMedia Release Supplement* provides late-breaking information on software bugs, workarounds, and other items not found in the other documentation sources.

## Changes to Backup for SmartMedia

Backup communicates with SmartMedia through a *virtual jukebox* resource. This term is used because SmartMedia takes over the management of the physical autochanger, relieving the Backup server or storage node from this task.

The virtual jukebox resource tracks the volumes that have been allocated for use by a Backup server. The virtual jukebox also maintains a list of volumes managed by SmartMedia that the Backup server can access. The number of slots in the virtual jukebox increases as volumes are allocated and decreases as volumes are deallocated.

The number of devices added to the virtual jukebox defines the maximum number of SmartMedia volumes that the Backup server can access simultaneously. This number is an upper boundary because devices managed by SmartMedia are shared, therefore they might not always be available.

The devices in the virtual jukebox resource are of type “logical.” The name of the device resource does not have to be a name typically used for tape devices for the operating system in question. Therefore, you might use any name for the device resource.

When Backup sends a request to SmartMedia to mount a volume, Backup allows SmartMedia to select the device. After the volume is mounted, Backup updates the device resource with the name of the device selected by SmartMedia. The device resource attribute’s logical name is used to record the name of the device selected by SmartMedia.

The `jb_config` command creates the virtual jukebox. After configuring the virtual jukebox, use the `nsrjb` command to access the volumes that are managed by SmartMedia. See Appendix B, “Command Line Reference Utilities,” on page page 261 or the `nsrjb` and `jb_config` man pages for detailed information on using these commands.

## The `jb_config` Command

The `jb_config` command configures a virtual jukebox for your Backup server after obtaining information about your network through a setup script. The virtual jukebox resource keeps track of the media allocated for use by the Backup server and managed by SmartMedia.

You must create a virtual jukebox resource for *each* Backup server that needs to access devices managed by SmartMedia. However, you do not need to create a virtual jukebox for a storage node. For example, if you are configuring a Backup server and storage node, you need to create a virtual jukebox for the server. If you are configuring two Backup servers, you need to create two virtual jukeboxes, one for each server.

The section “How to Create a Virtual Jukebox for Backup” on page 223 provides instructions for using the `jb_config` command. For more information and a complete list of `jb_config` options, refer to the `jb_config` man page.

**Caution –** Choose the “Install a SmartMedia Jukebox” option displayed by the `jb_config` command *only* on Backup servers that are installed with the SmartMedia software.

After the autochanger configuration is complete, you must run the `nsrcap` command to load a license for the autochanger you just configured. See the `nsrcap` man page for more information.

## The nsrjb Command

The nsrjb command allows Backup to access resources that are being managed by SmartMedia. Use the nsrjb command to allocate, label, load, unload, and deallocate the volumes managed by SmartMedia. For more information and a complete list of nsrjb options, refer to the nsrjb man page.

# How Backup and SmartMedia Interact

The SmartMedia server, DCP, and LCP components are organized as a set of cooperating processes. The SmartMedia server is a multithreaded process that accepts client connections and fulfills access requests from the Backup server or storage node by forwarding them to appropriate library control programs (LCP) and device control programs (DCP). The SmartMedia server maintains a catalog containing information about cartridges in the system and descriptions of authorized applications, libraries, and drives.

FIGURE 9-3 on page 217 illustrates how the main components of the SmartMedia and Backup software communicate with each other:

1. The Backup server or storage node sends a request to SmartMedia to mount a volume.
2. The SmartMedia server tells the LCP to load a cartridge into a drive.
3. The LCP communicates the instructions to the robotic library.
4. The library informs the LCP when the task is completed.
5. The LCP tells SmartMedia that the task has been completed.
6. The SmartMedia server tells the DCP to initialize the drive.
7. The DCP communicates the instructions to the drive.
8. The drive informs the DCP when it is ready.
9. The DCP tells the SmartMedia server that the drive is ready.
10. The SmartMedia server gives the Backup server or storage node the location of the volume.
11. The Backup server or storage node sends a command to the drive to read, write, or label the volume.

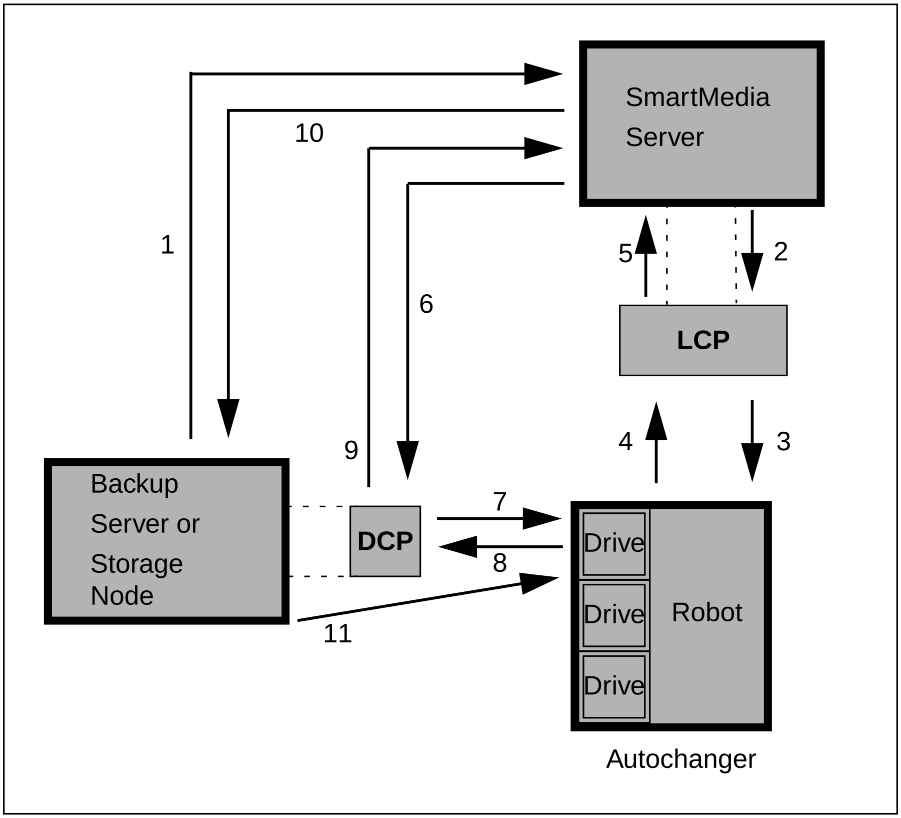


**FIGURE 9-3** SmartMedia Architecture

# Overview of Configuration Tasks

This section provides instructions for setting up SmartMedia for two basic configuration scenarios:

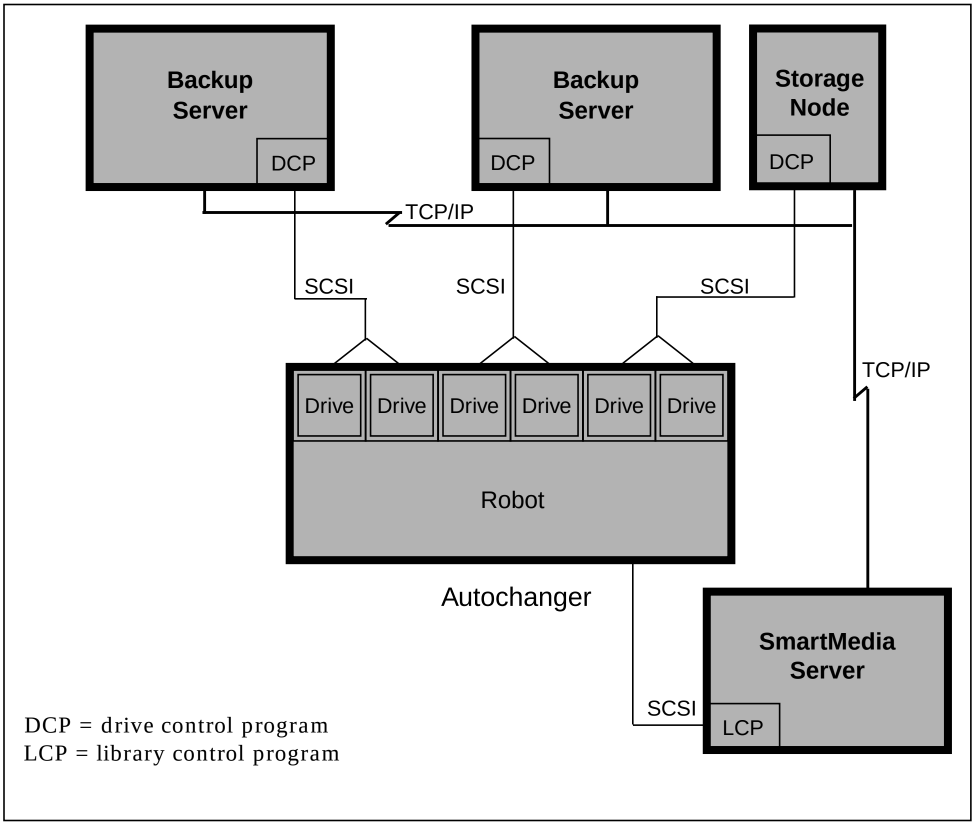
- A Backup server and storage node sharing an autochanger (see FIGURE 9-4 on page 218)
- Two Backup servers sharing an autochanger (see FIGURE 9-5 on page 219)

Most of the steps for the two configuration scenarios are the same. Regardless of the configuration of your network, you must complete two tasks in order to use Backup with SmartMedia:

1. Configure the SmartMedia server so that SmartMedia recognizes each Backup server and storage node.

2. Configure each Backup server with a virtual jukebox.

The difference in the configuration procedure occurs when you answer the prompts to the SmartMedia setup script for adding security keys to Backup. (This is described in detail in "How to Add Security Keys to SmartMedia" on page 220.)



DCP = drive control program
LCP = library control program

**FIGURE 9-4** Setting Up a Backup Server and Storage Node

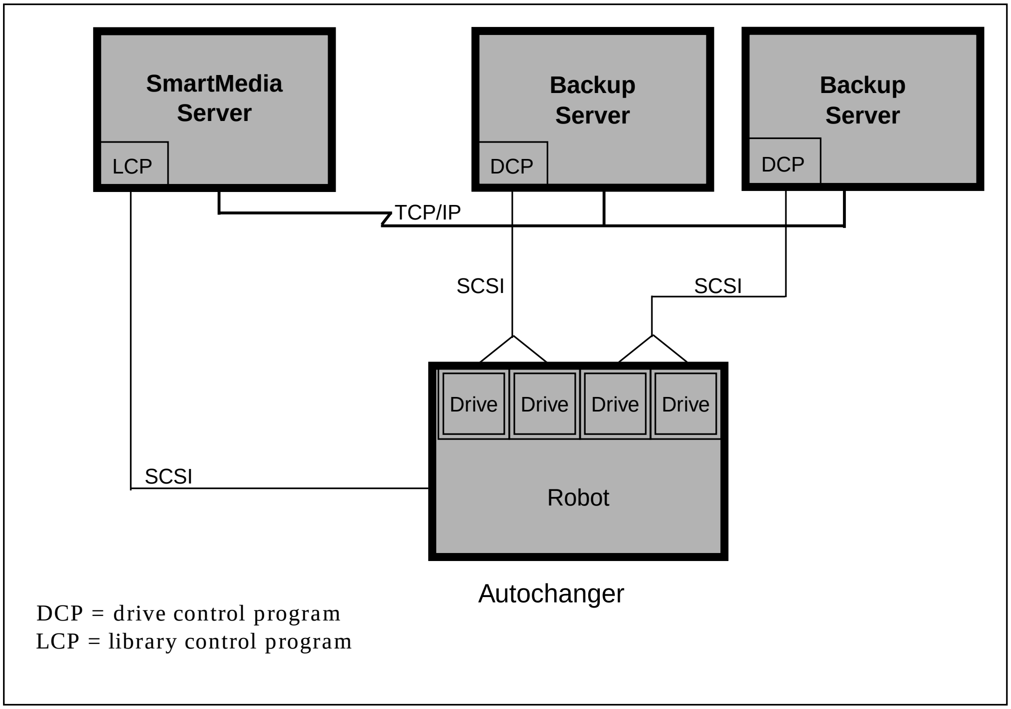


**FIGURE 9-5** Setting Up Two Backup Servers

Following is an overview of the tasks required to set up Backup for SmartMedia.

Refer to the *SmartMedia Administrator's Guide* for detailed instructions for Step 1 and Steps 3-8.

1. **Set up and configure the SmartMedia server, LCPs, and DCPs using the SmartMedia setup script.**

2. **Set up Backup as a client application of the SmartMedia server by adding security keys for Backup.**

   This allows the Backup server or storage node to use the libraries and drives managed by the SmartMedia server. See "How to Add Security Keys to SmartMedia" on page 220 and "Example: Adding Security Keys" on page 221.

3. **Start the SmartMedia server with the** `/etc/init.d/SmartMedia start` **command**.

4. **Use the** `ov_import` **command so the SmartMedia server recognizes the media within the autochanger.**

5. **Use the** `ov_cartgroup` **command to create cartridge groups as appropriate:**

a. **If the Backup servers and storage nodes require access to all the media, create *one* cartridge group.**

b. **If the Backup servers and storage nodes require separate groups of media, create two or more cartridge groups.**

6. **Use the** `ov_cartgroup` **command to create a cartridge group application for each application. This allows applications to access specified cartridge groups.**

7. **Use the** `ov_drivegroup` **command to create drive groups for all devices.**

8. **Use the** `ov_drivegroup` **command to create a drive group application for each application. This allows an application to access specified drive groups.**

9. **Use the** `jb_config` **command to create a virtual jukebox for *each* Backup server and storage node.**

   See "How to Create a Virtual Jukebox for Backup" on page 223 and "Example: Configuring a Virtual Jukebox" on page 225.

10. **Use the** `nsrjb` **command to allocate, label, load, unload, and deallocate the volumes managed by the SmartMedia server. Refer to the** `nsrjb` **man page for more information.**

## ▼ How to Add Security Keys to SmartMedia

When you set up the SmartMedia server, you need to add authorization security keys for each application that requires access to devices managed by SmartMedia. Adding security keys enables the SmartMedia server to recognize the application.

---

**Caution –** You must run the SmartMedia setup script for *each* Backup server and storage node that needs to access devices managed by SmartMedia. Run the script from the host computer of the SmartMedia server.

---

Before you begin the script, review this section for an explanation of the setup prompts. The *client application* referenced by the prompts is Backup.

1. **What is the hostname of the client application?**

   When setting up Backup, you must enter the host name of the system where the *devices are attached*. Note that this might not be the same location as the client application.

   For all other applications *except* for Backup, follow the instructions in the *SmartMedia Administrator's Guide* and enter the hostname of the client application.

2. **What is the name of the client application?**

Provide your response for this prompt in the format backup@*hostname*

The value for *host-name* depends on whether you are configuring a Backup server and storage node or two Backup servers.

For a Backup server and storage node:

a. **When running the setup script for the Backup server, use the hostname of the server where Backup is located (for example,** backup@*serv0*, **where** *serv0* **is the name of the host).**

b. **When running the setup script for the storage node, enter the same response as provided for the Backup server (for example,** backup@*serv0***).**

For two Backup servers:

- When running the setup script for the first Backup server, use the first server as the hostname, (for example, backup@*Serv1*, where *Serv1* is the name of the server).
- When running the setup script for the second Backup server, use the second server as the hostname, (for example, backup@*serv2*, where *serv2* is the name of the server).

See Table 9-1 on page 223 for sample responses to this prompt.

3. **What is the instance name of the client application?**

   When using this script for Backup, you must provide the same response to this question as provided for Question 1—that is, the system where the *devices are attached*.

4. **Does the client application have administrator privileges? (Default: No)**

   When setting up Backup, you must answer No to this question. Otherwise, the SmartMedia server will not recognize Backup as a client application.

5. **What is the authorization security key for the client application?**

   Each client has a name or *security key* that is used to identify the client. If you do not want to provide a unique name, type the default none at the prompt. Make sure that your response to this prompt is the same for the Backup server and the storage node.

## Example: Adding Security Keys

To add security keys to the SmartMedia server for Backup, complete the steps that follow. For an explanation of the setup prompts, see “How to Add Security Keys to SmartMedia” on page 220.

**Caution –** You must run this setup script for each Backup server and storage node that needs to access the devices SmartMedia manages. Run the setup script from the host system of the SmartMedia server.

1. **Become superuser on the host system of the SmartMedia server with the following command:**

```
# su -
```

2. **Change to the directory where you installed SmartMedia using the following command:**

```
# cd /opt/SmartMedia
```

3. **Start the configuration script:**

```
# setup
```

The SmartMedia Configuration Menu appears as follows:

```
SmartMedia Configuration Menu
1. Configure the SmartMedia Server
2. Configure Admin and User Commands

3. Configure a new LCP
4. Modify an existing LCP
5. Deconfigure an existing LCP

6. Configure a new DCP
7. Modify an existing DCP
8. Deconfigure an existing DCP

99. Exit.
```

4. **At the prompt, enter `1`. The SmartMedia Server Setup Menu appears as follows:**

```
SmartMedia Server Setup Menu
1. Configure the Server Parameters
2. Add New DCP Keys To The Server
3. Add New LCP Keys To The Server
4. Add New Admin CLI Tools Keys To The Server
5. Add New Client Application Keys To The Server
98. Return to Main Menu.
99. Exit.
```

5. **At the prompt, enter `5`. The Security Key for Client Applications script begins.**

6. **Enter your response for each prompt as shown below. Table 9-1 provides sample responses based on your configuration.**

```
What is the host name of the Client Application?
(default: Serv1):
Enter the name of the Client Application (default: *):
What is the instance name of this Client Application? (default: *):
Does the client application have administrator privileges?
(default: no):
Enter the security key of this client application
(default: none):
```

**TABLE 9-1** Sample Responses

| | **Backup Server & Storage Node** | | **Two Backup Servers** | |
|---|---|---|---|---|
| **Script Prompt** | **Server** | **Storage Node** | **Server1** | **Server2** |
| Application hostname | `Serv0` | `SNode` | `Serv1` | `Serv2` |
| Application name | `Backup@Serv0` | `Backup@Serv0` | `Backup@Serv1` | `Backup@Serv2` |
| Instance name | `Serv0` | `SNode` | `Serv1` | `Serv2` |
| Administrator privileges | `No` | `No` | `No` | `No` |
| Security key | `None` | `None` | `None` | `None` |

The script returns to the SmartMedia Server Setup Menu.

7. **Repeat Steps 5 and 6 for your storage node.**

8. **Enter** 98 **to return to the SmartMedia Main Menu or enter** 99 **to exit the setup script.**

## ▼ How to Create a Virtual Jukebox for Backup

Backup communicates with SmartMedia through a virtual jukebox. You need to use the `jb_config` command to create a virtual jukebox for *each* Backup server and storage node.

Before you begin the `jb_config` script, review the questions that follow. Table 9-2 on page 225 includes sample responses for the setup script and has additional space for you to record your own responses.

1. **What name would you like to assign to the SmartMedia jukebox?**

   The name of the device resource does not have to be a name typically used for tape devices. Therefore, you might use any name for the device resource.

2. **What is the host name of the SmartMedia server? [Default:** `host-name`**]**

   The SmartMedia setup script also asks for this information. Make sure that your entries for the SmartMedia setup and `jb_config` scripts are the same for this prompt.

3. **What is the port number of the SmartMedia server? {Default:** 44444**]**

   The SmartMedia setup script also asks for this information. Make sure that your entries for the SmartMedia setup and `jb_config` scripts are the same for this prompt.

   When selecting a port number, do not use zero. Zero is a special number that tells the system to use any available port number. In addition, you might want to avoid using port numbers 1–1024 because these numbers are reserved for privileged processes running as root.

   The default number 44444 was selected to avoid using a port that is already designated for another process. Otherwise, this number has no significance.

4. **How many devices do you have to configure (1 to 64)? [Default:** 4**]**

   Enter a number from 1 to 64 for this question. Your response defines the number of logical devices added to the virtual jukebox, and determines the maximum number of SmartMedia volumes that the Backup server or storage node can simultaneously access. This number is an upper boundary because devices managed by SmartMedia are shared, and therefore might not always be available.

Only enter the *actual* number of devices that you plan to configure. There is no benefit to entering a number greater than the actual quantity of physical devices.

5. **What is the hostname of the storage node for logical device 1? [Default:** *host-name***]**

6. **What is the name of the logical device?**

   Enter any name for the logical resource.

   Based on your response to Question 4, the script asks you the previous two questions for *each* device added to the virtual jukebox. Thus, if you selected the default 4 for Question 4, the script will repeat Questions 5 and 6 four times.

7. **What is the name of the application? [Default:** `Backup@`*host-name***]**

   The SmartMedia setup script also asks for this information. Make sure that your entries for the SmartMedia setup and `jb_config` scripts are the same for this prompt.

   Include the hostname of the Backup server when you respond to this prompt, using the format:

   `Backup@`*Backup-server-host-name.*

8. **What is the security key for this application? [Default:** `none`**]**

   The SmartMedia setup script also asks for this information. Make sure that your entries for the SmartMedia setup and `jb_config` scripts are the same for this prompt.

   Each client application has a name or authorization *security key* that is used to identify the client. If you do not want to provide a unique name, select the default `none` at the prompt.

**TABLE 9-2** Virtual Jukebox Information

| Script Question | Sample Responses | | | Your Response |
|---|---|---|---|---|
| SmartMedia Server Hostname[1] | *Serv0* | *SNode* | *Serv1* | |
| SmartMedia Server Port Number[1] | *44444* | *44444* | *44444* | |
| Device Quantity (1-64) | *2* | *2* | *2* | |
| Storage Node Hostname | `Serv0`<br>`Serv0` | `SNode`<br>`SNode` | `Serv1`<br>`Serv1` | |

**TABLE 9-2** Virtual Jukebox Information *(Continued)*

| Script Question | Sample Responses | | | Your Response |
|---|---|---|---|---|
| Logical Device Name | dev1<br>dev2 | dev3<br>dev4 | dev5<br>dev6 | |
| Application Name[1] | Backup@Serv0 | Backup@Serv0 | Backup@Serv1 | |
| Key[1] | none | none | none | |

[1] This also appears in the SmartMedia server setup script. Make sure your responses to these items in the `jb_config` script match your entries for the SmartMedia setup script.

## Example: Configuring a Virtual Jukebox

To create a virtual jukebox, complete the steps that follow. See "How to Create a Virtual Jukebox for Backup" on page 223 for an explanation of the setup prompts.

**Caution –** You must run the `jb_config` script for each Backup server and storage node that needs to access devices managed by SmartMedia. Usually, the `jb_config` command should be executed on the host computer of the Backup server. However, you can run it from any computer that has root privileges to the Backup server.

1. **Become superuser with the following command:**

```
# su -
```

2. **Enter the** `jb_config` **command at the prompt to initiate the virtual jukebox configuration script:**

```
# jb_config
```

3. **Enter your response for each prompt:**

```
Name you would like to assign to the SmartMedia jukebox? jukebox1
Name of host machine for SmartMedia server? [Serv1] [return]
Port number of SmartMedia server? [44444] [return]
How many devices are to be configured (1 to 64)? [4] 2
Enter name of storage node for logical device 1: [Serv1] [return]
Enter name of logical device 1: dev1
Enter name of storage node for logical device 2: ? [Serv1] [return]
Enter name of logical device 2: dev2
Enter application name defined in SmartMedia for Backup?
[Backup@Serv1] [return]
Enter application key defined in SmartMedia for Backup? [<none>]
[return]
The barcode reader is enabled and volume labels are set to match
barcode labels.
Jukebox has been added successfully.
```

4. **Repeat the** `jb_config` **script for each Backup server and storage node that needs to access the devices managed by SmartMedia.**

CHAPTER **10**

# SNMP Module

This chapter provides instructions to configure and use the Simple Network Management Protocol (SNMP) Module from your SunNet Manager or HP OpenView Network Node Management window. The SNMP Module is a separate optional add-on module for Backup. This chapter addresses the following topics:

- What the Backup SNMP Module Provides
- SNMP Notification Configuration
- Backup SNMP Defaults
- SunNet Manager
- HP OpenView Network Node Manager

## Overview

SNMP is a protocol for network transactions that specifies the transfer of structured management information between SNMP managers and agents.

An SNMP manager, residing on a workstation that can be the Backup server or a client of the Backup server, issues queries to the SNMP agent to gather information about status, configuration, and performance. The SNMP agent, which resides on the Backup server, responds to the queries issued by the SNMP manager and generates activity reports. In addition to responding to SNMP queries, the Backup SNMP module transmits unsolicited reports, or traps, to the SNMP manager when events for which a notification is set up on the Backup server occurs.

The Backup SNMP Module allows communication of Backup event notifications to network management stations that comply with the SNMP standard through the SNMP *trap* mechanism. An SNMP trap is an unsolicited notification sent from the SNMP agent to the network manager’s event manager.

For detailed explanations of SNMP operation, refer to your network management software documentation.

# What the Backup SNMP Module Provides

The Backup SNMP option enables you to use using network management software to:

- Receive Backup event notifications and status information through the network management window
- Launch the Backup Administrator window from the Tools menu in the SunNet Manager window or the Misc menu in the HP OpenView Network Node Manager window

The `nsrtrap` SNMP daemon must be running on both the Backup server and the network management station to send and receive SNMP traps.

# SNMP Notification Configuration

Icons representing your Backup servers are displayed on your network management console. From this console, using your network management software, you can:

- Configure the manner of event trap notifications (for example, flashing icon or color change)
- Create new SNMP notification schemes, through the Backup server administrator program, with different priorities and events
- Track pending, alert, and other configured messages

## Customized Backup Notifications

You can customize Backup notifications to set priorities, specify which types of events send traps, and specify which traps are sent to specific destinations. When you select the Details option on the View menu, the Notifications window displays check boxes that represent the events and priorities available to choose from. Highlight any number of check boxes to customize your event, provide a unique name for the notification, and save your customized notifications. See “Notifications Resource” on page 24 for more information on Backup notifications and associated priority values.

# Backup SNMP Defaults

The following table provides the default SNMP information specific to Backup.

| SNMP Parameter | Backup Default |
|---|---|
| host-name | network management station name |
| community | public |
| enterprise object ID | 160 (.1.3.6.1.4.1.160) |
| trap-type | 1 |

# SunNet Manager

The SunNet Manager provides services for capturing SNMP traps on the Solaris platform.

## ▼ How to Configure SunNet Manager to Receive Backup Notifications

The following modifications to the /var/opt/snm/snmp.traps file formats the trap messages and notifications from the Backup server to make them more readable:

1. **Use your favorite text editor to open the** snmp.traps **file for editing.**
2. **Add the following lines to the end of the file:**

```
enterprise 1.3.6.1.4.1.160
1 Backup_Trap
Using SunNet Manager
```

3. **Start the SunNet Manager program.**

   When you start SunNet Manager, the Home window of the SunNet Manager Console is displayed.

4. **Add the Backup option to the SunNet Manager Tools menu.**

## ▼ How to Add the Backup Option to the Tools List

1. **Select Customize from the Tools menu; the SunNet Manager Console: Custom Tools window is displayed.**

2. **Enter the following text in the Tool Name field to add Backup to the Tools List:**

```
Backup
```

If your network management station is not running Backup, or if you need to access more than one Backup server, specify the machine name of your Backup servers in the Tool Name field:

```
Backup_server-name
```

If you are adding more than one Backup server to the Tools List, be sure to give each one a unique name.

3. **Enter the command for starting the Backup administration program in the Command field:**

```
# nwadmin
```

If your network management station is not the Backup server, include the `-s` option and specify the machine name of your Backup server.

```
# nwadmin -s server-name
```

4. **Click the Add button; the Backup option is displayed in the Tools List.**

You can create customized notifications, using the Backup server administration program, to send and trap customized Backup notifications to your SunNet Manager Console.

## ▼ How to Configure the Backup Server to Send Notifications

Use the following instructions to configure your Backup server to send event and trap notifications to SunNet Manager:

1. **Double-click the icon in the Home window of the SunNet Manager Console to gain access to servers and clients on your network.**
2. **Expand the network to display and select the Backup server to configure.**
3. **Select Backup in the Tools menu; the Backup administration window is displayed.**
4. **Select Notifications from the Customize menu.**
5. **Create a new notification, specifying the `nsrtrap` command for the Action to follow when the event is triggered.**
6. **Save and apply your changes to implement your notification.**

   The online help for the Notifications window contains detailed instructions on how to create a new notification for Backup events. Refer to the `nsrtrap(1m)` man page for more information about creating Backup SNMP notifications.

# HP OpenView Network Node Manager

The HP OpenView Network Node Manager provides services for capturing SNMP traps on the HP-UX platform.

## ▼ How to Configure HP OpenView Network Node Manager

The following modifications to the `trapd.conf` file formats the trap messages and notifications from the Backup server to make them more readable. For HP OpenView release 3.x, the `trapd.conf` file is in `/usr/OV/conf/C`. For HP OpenView release 4.x, the `trapd.conf` file is in `/etc/opt/OV/share/conf/C`.

1. **Use your favorite text editor to open `trapd.conf` for editing.**

2. **Add the following line to the “enterprises” section of the file:**

```
Sun {.1.3.6.1.4.1.160}
```

3. **Add the following lines to the end of the file:**

```
Backup_Trap {.1.3.6.1.4.1.160} 6 1 A “Application Alert Events” 4
$A: $1
SDESC
This event is sent by a Backup server when configured SNMP Backup
events occur.
EDESC
```

These lines configure the node manager to log events in the following format: *hostname:* `Backup-event-`*string*. The category specified is Application Alert Events, and the number 4 indicates the severity is set to “Major.” The $A parameter represents the host sending the Backup trap, and the $1 parameter represents the trap message.

## Configuring Event Notifications

Start HP OpenView in your usual manner. Two windows are displayed: the HP OpenView Network Node Manager window and the Event Categories menu.

In the Event Categories window, select the events for which you want to view notifications:

- Error Events
- Threshold Events
- Status Events
- Configuration Events
- Application Alert Events
- All Events

## ▼ How to Add the Backup Option

Use the following instructions to add the Backup option to the HP OpenView Misc menu:

1. **Become root on the system running HP OpenView.**

2. **Change directories by entering:**

```
# cd /usr/OV/registration/C
```

3. **Using your favorite text editor, create a file named** `nsrapp`**.**
4. **Add the following lines to** `nsrapp`**:**

```
Application “Backup”
{
               MenuBar “Misc” {
               “Backup” f.action NwAdmin;
               }
               Action NwAdmin {
               SelectionRule isNode;
               MinSelected 1;
               MaxSelected 1;
               Command “xterm -title ${OVwSelection1} -e \
               rsh ${OVwSelection1} \
               nwadmin -s ${OVwSelection1} -display \
               <display-workstation>:0”;
               NameField “IP Hostname”;
               }
             }
```

You can replace the string following Command with a shell script or command to execute when you select the Backup option from the HP OpenView Misc menu. For the example shown, replace *display-workstation* with the host name where you want Backup displayed, usually the system running HP OpenView.

## ▼ How to Send Notifications to the HP OpenView Console

Use the following instructions to configure your Backup server to send event and trap notifications to the HP OpenView console:

1. **Start the Backup administration program.**
2. **Change to the server you plan to configure.**
3. **Select Notifications from the Customize menu.**
4. **Create a new notification, specifying the** `nsrtrap` **command for the Action to follow when the event is triggered.**
5. **Save and apply your changes to implement your notification.**

CHAPTER 11

# Silos

This chapter describes the Silo Support Module that you can use with Backup NetWork Edition or Backup Power Edition. It also provides information about how to enable and use a silo with Backup. The following topics are addressed in this chapter:

- Silo Installation and Configuration
- Media Management in a Silo
- Silo Device Management

## Overview

A silo is a peripheral machine that typically contains many storage devices. Silos are controlled by silo management software, which is provided by the silo vendor and installed on a server. The silo server *cannot* be the same computer as the Backup server.

The silo and devices in the silo can be shared among many applications, systems, and platforms. Like autochangers, silos make data and media operations more automatic. Silos can load, change, and manage volumes and clean the devices automatically.

### How Backup Interacts With a Silo

A Backup server or storage node acts as a client of the silo management software. Backup communicates with the silo through the Silo Tape Library Interface (STLI) library.

To access the volumes and devices in a silo, Backup sends a request to the silo management software, in the form of an STLI library call. For example, to mount a volume in a silo device, the Backup daemon sends a request to the silo management software to mount the volume into a particular device in the silo. The silo server responds to the request and mounts the volume in the silo. For further details on this process, refer to the `stli` man page.

The silo management software controls many of the operations that Backup controls with an autochanger. For example, the silo management software keeps track of the slot where each silo volume resides and usually also controls the deposit and withdrawal of volumes and automated cleaning of silo devices.

## Silo Installation and Configuration

To use a silo with Backup, follow these steps:

1. **Install the silo management software on the silo server.**

2. **Do not install the STLI library on the following models, because all the necessary software was installed when you installed Backup:**

   - StorageTek on Solaris, AIX, HP-UX, and DYNIX/ptx
   - EMASS/Grau on Solaris, AIX, HP-UX, and Windows NT
   - IBM 3494 on Solaris and AIX

   For other silo models, install the STLI library on the Backup server or storage node that uses the silo. Follow the instructions provided by the silo vendor.

3. **Ensure that the Backup server or storage node is properly connected to the media devices in the silo that Backup uses.**

4. **Run the `jb_config` program to configure the devices in the silo for Backup to use. See "How to Configure a Silo" on page 238 for instructions.**

5. **Enable the Silo Support Module using the instructions on your Silo Support Module enabler certificate.**

6. **Register and authorize the Silo Support Module. See "How to Register and Authorize Your Software" on page 13 for instructions.**

## ▼ How to Configure a Silo

Use the `jb_config` program to configure the silo. The program prompts you to enter the following information:

- Type of silo.
- Name of the silo. Enter any alphanumeric string.
- Hostname of the silo server (StorageTek and EMASS/Grau).
- Symbolic name of the silo, as defined in the `/etc/ibmatl.conf` file (IBM).
- Whether to enable automatic device cleaning. Enter `no`. The silo management software controls device cleaning in silos.
- Pathname of the STLI library software. Enter the full path on the Backup server or storage node where the silo interface library software resides.
- Number of devices to configure.

For each device you configure, the program prompts you for the following information:

- Pathname of the device.
- Media type of the device (if the device is not already configured).
- Silo name of the device. See "Silo Device-Naming Conventions" on page 241 for information about the silo name to enter for each device.

The following example was created on a Backup server for Solaris. The output of the `jb_config` program varies slightly, depending on the operating system and the type of silo.

```
# jb_config
1) Install a SmartMedia Jukebox.
2) Install an Autodetected SCSI Jukebox.
3) Install an SJI Jukebox.
4) Install an RLM Jukebox.
5) Install an STL Silo.
What kind of Jukebox are you installing? [1] 5
Supported Silo types for this system are:
        1) ACSLS Silo(StorageTek)
        2) DAS Silo(Emass/Grau)
        3) 3494 Silo(IBM)
Enter the number corresponding to the type of silo you are
installing:
Choice? 1
Installing a StorageTek Silo.
Name you would like to assign to the Silo device? stk_silo
Name of the host running the ACSLS software? [] expo1
Pathname of the STL library for the ACSLS silo? [/usr/lib/nsr/
libstlstk.so] [Return]
Do you want automated device cleaning support enabled? (yes/no) n

How many devices are to be configured for this silo (1 to 64)? [4] 1
Enter pathname of media drive 1: ? /dev/rmt/0mbn
This media device has not been configured yet. Please select a
media device type for /dev/rmt/0mbn.
                a) himt
                b) qic
                c) 4mm
                d) 8mm
                e) 8mm 5GB
                f) 3480
                g) dlt
                h) optical
Choice? h
Enter corresponding silo name of media drive 1: ? 0,0,2,0
Since this is a silo, the barcode reader is enabled, and volume
labels are set to match barcode labels
Jukebox has been added successfully
```

You can view the results of your silo configuration in the Jukeboxes resource in the Backup administration program (`nwadmin`) or in the `nsradmin` program. Refer to the online help or the `nsr_jukebox` man page for details on the attributes in the Jukeboxes resource.

# Silo Device-Naming Conventions

The `jb_config` program prompts you for the silo name of the storage devices, which is the name of a device in the silo. The silo name is the name that the Silo Management Software uses to refer to that device. Depending on the type of silo you have, the device name can take several forms. This section describes the naming conventions of the currently supported silos.

## StorageTek

The Storage Tek (STK) silo management software (either a program called ACSLS that runs on a UNIX system or a program called Library Attach that runs on an MVS system) names devices according to a coordinate system based on the physical location of the devices in the silo.

For tape drives, the name consists of four digits separated by commas. The first digit refers to the automated cartridge system (ACS) with which the drive is associated. The second digit refers to the library storage module (LSM) in which the drive is located. The third and fourth digits refer to the panel and slot location in which the drive is located. A typical name for an STK drive looks like 1,0,1,0.

You must ask the silo administrator the drive names for the devices that Backup can use. There is no method to find this information from the Backup system. To connect to more than one drive, find out the SCSI IDs for each drive and properly match the IDs to the silo names. If you accidentally swap operating system device names and silo names, you can only mount and unmount volumes; you cannot read or write to the volumes after they are mounted. To reconfigure the device names properly, use the Backup administration program to change the order of the device names in the STL Device Names attribute of the Jukeboxes resource.

## IBM 3494

The silo management software for the IBM 3494 names devices with an 8-digit number to identify the 3590 drives in the silo. Use the appropriate utility to obtain the device names, as follows:

- On an AIX system, the `jb_config` program gets the name of the device from the device driver and displays the device name as the default value.
- On a Solaris system, you must use the IBM supplied `mtlib` command (`mtlib -l` *library-name* `-D`) to determine the names of all the devices in the 3494. Either ask the silo administrator which device is reserved for Backup, or test to decide which silo drive name matches with each Solaris device name.

### EMASS/Grau

The silo management software for the EMASS (in North America) or Grau silos is a program called DAS. DAS acts as a front end for the silo control program called AMU. When the silo is configured, the silo administrator gives each drive a symbolic name. The symbolic name can be any alphanumeric string.

To set up DAS to work with Backup, follow these steps:

1. **Ask the silo administrator to configure DAS to accept commands from your Backup server or storage node computer.**
2. **Ask the silo administrator to either:**
   - Use the `dasadmin allocd` command to allocate one or more devices to the Backup server or storage node.
   - Configure your Backup server or storage node as an administrator, so you can enter the `dasadmin allocd` command to allocate devices from your Backup server or storage node computer.

To find the names assigned to the devices in the silo, you can use a utility called `dasadmin` that is supplied with the Backup software.

1. **Set three environment variables:**
   - DAS_SERVER, the hostname of the silo management server, which runs DAS
   - DAS_CLIENT, the hostname of the Backup server or storage node
   - ACI_MEDIA_TYPE, one of the following values: 3480, CD_THICK, CD_THIN, DECDLT, 8MM, 4MM, D2, VHS, 3590, CD, TRAVAN, DTF, BETACAM, AUDIO_TAPE, or DAS_MEDIUM, same value as ACI_MEDIA_TYPE
2. **Issue the following command to see a list of drives and the hostnames to which they are allocated:**

```
# dasadmin ld
```

## Media Management in a Silo

Because more than one software application can use a silo, media management in a silo requires extra operations to protect the volumes used by other programs from being overwritten by Backup.

The tasks described in this section deal with how volumes are specified for Backup to use, how volumes are mounted in the devices, and how volumes are identified and counted in the silo.

## What Slot Numbers Mean in a Silo

In an autochanger, Backup specifies many of the functions by a slot number; silos use this same idea. In an autochanger, there is a fixed number of slots; Backup uses the slot number to refer to the physical location of a volume. However, a silo has a variable number of slots, starting at zero when you first configure it and limited by the silo license you purchased. The fundamental identifier of a silo volume is its barcode, which is often called a "volser" in silo documentation. The value for volser never changes over the life of a particular volume.

When the `nsrjb` command displays a list of the contents of a silo, it also displays a slot number. You can use the slot number to specify which volumes to mount, unmount, label, or inventory. Volumes are not always assigned the same slot number in the silo. The slot numbers in the silo are assigned dynamically, based on the sorted order of the barcodes that have been allocated. If you allocate more barcodes that fall earlier in the sort sequence, the slot numbers of all the volumes that come later in the sequence change.

Because the slot number is not a perfect identifier for silo volume, operations that might change the slot number cannot accept slot numbers as arguments. For example, you cannot deallocate volumes based on slot numbers, because this operation can change the slot numbers of volumes being deallocated.

## ▼ How to Use the CAP to Deposit and Withdraw Volumes in a Silo

A Cartridge Access Port (CAP) enables you to deposit and withdraw volumes in a silo without opening the door to the silo. The CAP is useful because you can add (deposit) and remove (withdraw) volumes in a silo without having to reinventory the entire silo. When you use the CAP to add or remove volumes, Backup does not automatically take inventory, read barcode labels, or locate empty slots in the silo. Use the silo inventory feature and Jukeboxes resource for these tasks.

You can use Backup commands or the silo management software to control the CAP on the currently supported silos to deposit and withdraw volumes in a silo. It is often more efficient to use the silo management software, especially to deposit or withdraw a large number of volumes.

The Backup command to allocate and deposit volumes is the `nsrjb -a -T`*xxxx* `-d` command. The Backup command to deallocate and withdraw volumes is the `nsrjb -x -T`*xxxx* `-w` command. The deposit and withdraw functions are not available in the Backup administration program GUI.

On some silos (IBM 3494 and StorageTek silos when the CAP is set to automatic mode), the silo management software inserts volumes automatically. You cannot use Backup to insert volumes.

On StorageTek silos, due to differences between the internal operations of Backup and the StorageTek silo management software, Backup can only withdraw one volume at a time. You must physically remove the volume from the silo's CAP before you can withdraw any more volumes. On EMASS/Grau silos, Backup can control both the deposit and withdraw functions.

## ▼ How to Allocate Volumes in a Silo

When you allocate volumes, you tell Backup which volumes it can use. Because more than one software application can use a silo, it is possible that a different application could read or write to volumes that belong to Backup. To prevent this problem, most silo management programs include methods of limiting access to volumes based on the hostname of the computer on which Backup and the other programs run. Backup does not provide any method for setting up this sort of protection; it must be configured by the silo management program.

When you allocate a volume, Backup queries the silo management software to verify that the volume you requested exists. If the volume exists, the volume is allocated to Backup. If the volume does not exist, the following message is displayed:

```
barcode XXXXXX is not present in the silo and was not added
```

If you are allocating a range of volumes, the allocation continues after displaying the message. The message is informational and does not indicate a Backup error.

To allocate a silo volume, use either of the following methods:

- The Add Library Volumes dialog box in the Backup administration program. Refer to the online help for instructions.
- The `nsrjb -a -T` *barcode* command. Refer to the `nsrjb` man page for further information on this `nsrjb` command, and refer to the `stli` man page for the proper format of barcode templates. You can use only a single barcode identifier or template after the `-T` option, but you can use the `-T` option more than once with the same `nsrjb` command.

To deposit volumes into a silo and then allocate them (on silos that require manual depositing, such as EMASS/Grau), place the volumes in the insert area, then issue the following command:

```
# nsrjb -a -Txxxx -d
```

On StorageTek and IBM silos, the silo management software deposits volumes automatically.

## ▼ How to Mount and Unmount Volumes in a Silo

You must mount a volume before you read or write data on it. Volumes are mounted in a device in the silo by the robotic mechanism. To mount a volume in a silo device, you can use either the Backup administration program or the `nsrjb -l` command. When you mount a volume, you must specify the volume, slot, or barcode.

You must dismount a volume before you can inventory the volumes in a silo or deallocate the volume from a Backup silo. To unmount a volume, you can use either the Backup administration program or the `nsrjb -u` command.

To specify a barcode identifier or template for the volumes, you can use the `-T` option with either `nsrjb` command.

## ▼ How to Label Volumes in a Silo

Labels tell Backup the pool to which the volume belongs and what type of data the volume should contain. (For more information on volume labels, see “Labeling Storage Volumes” on page 82.) Backup cannot write data to a volume until you label the volume.

To label a volume in a silo, use either the Backup administration program or the `nsrjb -L` command.

Backup labels for volumes in a silo include both a regular Backup volume label (written on the media of the volume) and a silo barcode identifier. The volume label is usually based on the volume pool’s label template. The barcode identifier is written on a physical label on the outside of the volume, which the barcode reader in the silo can scan during inventory.

By default, the use of barcodes with matching barcode labels and Backup volume labels are both enabled for a silo. You can change the Match Barcode Labels attribute, but you should not set the Barcode Reader attribute to No. When you set the Match Barcode Labels attribute and the Barcode Reader attribute to Yes, the internal volume label Backup writes on the media of each volume matches the barcode label on the outside of the volume. When the labels match, it is easier to track volumes, but Backup does not require the internal and external labels to match.

With most silo management software, you can use unlabeled volumes. The management software assigns a “virtual” barcode label to those volumes. Although you can use volumes without barcodes, it is very difficult to maintain integrity. This is because after you remove the volume from the silo, the information about the virtual barcode is lost. You can reinsert any volume without a barcode into the silo under a virtual barcode that Backup (or another application) associates with some of your data.

## ▼ How to Deallocate Volumes in a Silo

If you no longer need a volume in a silo for Backup, you can deallocate the volume. Deallocation is basically the same operation as removing a volume from an autochanger. Although the robotic mechanism cannot load the volume, the entries in the Backup media database remain intact. If you allocate the volume again later, Backup can retrieve the data from it.

Use deallocation when your silo license limits the number of slots you can use or when you move data offsite for safer storage. In the case in which the license limits the number of slots, you might be able to leave the volumes in the silo, so you can easily reallocate the volumes when you need to access the data on them.

The allocation operation is not automatic. You must manually allocate the volumes again and reinventory them to let Backup access the data. In the case of removing the volume from the silo for off-site storage, you can either deallocate it with Backup and then use the silo management software to eject it from the silo, or you can perform both operations at the same time from the command line with the `nsrjb -x -T`*xxxx* `-w` command.

---

**Caution –** Currently STK silos can eject only one volume at a time. The silo operator must remove each volume before another `nsrjb -x -w` command can be issued. If you deallocate and withdraw multiple volumes, they are all deallocated, but only the first is ejected. This limitation does not exist on EMASS/Grau or IBM 3494 silos.

---

To deallocate a silo volume, follow these steps:

1. **Unmount the volume from the device.**

2. **Use either the Backup administration program or the** `nsrjb -x -T` **barcode command to deallocate the volume.**

Refer to the online help for information about how to deallocate a silo volume using the Backup administration program. Refer to the nsrjb man page for further information on the nsrjb command. Refer to the stli man page for the correct format of barcode templates.

### ▼ How to Inventory a Silo

You should take inventory of the volumes in a silo to make sure that the mapping between slot number and volume name is correct or to reconcile the volumes in a silo with the list of volumes in the Backup media database.

The slot number of a silo volume is not a numbered slot inside the silo, as it is in an autochanger. The slot number of a silo volume is the number of the volume's position in the list of volumes in a silo. You can view the slot number for each silo volume in the Backup administration program in the Jukebox Mounting dialog box.

Use the Backup administration program or enter the `nsrjb -I` command to inventory a silo. Backup examines all the volumes in the silo and compares the new list of volumes to the Backup media database. Then Backup produces a message listing any volumes located in the silo but not in the media database.

When Backup inventories a silo, the silo’s barcode label reader reads the barcode labels on the outside of each volume. When a barcode matches an entry in the Backup media database, Backup does not need to load the volume into the device and read it, and the inventory proceeds very rapidly. However, if Backup reads a barcode that does not match any of the entries in the media database, the volume must be mounted and read for a proper inventory.

You can inventory a range of slots or barcode labels if you are pressed for time and do not want to inventory all the volumes in your silo.

## Silo Device Management

This section also discusses the following device management tasks:

- How to define a device to Backup
- How to specify whether Backup has exclusive rights to a device or whether the device is shared among applications
- How to clean devices in a silo

You can use the Backup administration program or the nsradmin program to do many of the device management tasks. For some tasks, you must enter a command at the system prompt.

## ▼ How to Define a Silo Device

If you add a new device to the silo and you want Backup to recognize and use the device, you can use either of two methods to reconfigure your silo to access the new device.

- Method 1:

1. **Become root on the Backup server or storage node computer that uses the silo.**
2. **Delete the silo definition. Use either the Backup administration program or the** `nsradmin` **program.**
3. **Run the** `jb_config` **program to reconfigure the silo. Use the same name for the silo as before, and add the additional device.**

- Method 2:

1. **Become root on the Backup server or storage node computer that uses the silo.**
2. **Enter the following to edit the silo definition.**

```
# nsradmin - c
```

3. **Select Dynamic and Hidden from the Options menu.**
4. **Select** `nsr jukebox`**.**
5. **Select Edit.**
6. **Add the new device name to the Devices attribute. You must enter a comma and a space between the device names.**
7. **In the Number Devices attribute, increase the number of devices by one.**
8. **Add a null string with a comma (,““) to each of the following attributes, before the final semi-colon:**
9. **Loaded Volumes**
10. **Loaded Barcodes**
11. **Loaded Slots**
12. **Add the name of the new device in the STL Device Names attribute. Surround the device name with quotation marks.**
13. **Add the following entry to the Allocated Devices attribute:** `, No`

The following examples show selected `nsradmin` attributes before and after you add a device:

Before:

```
devices: /op1;
number devices: 1;
allocated devices: No;
loaded volumes: ““;
loaded bar codes: ““;
loaded slots: ““;
STL device names: “0,0,2,0”;
```

After:

```
devices: /op1, /op2;
number devices: 2;
allocated devices: No, No;
loaded volumes: ““, ““;
loaded bar codes: ““, ““;
loaded slots: ““, ““;
STL device names: “0,0,2,0”, “0,0,2,1”;
```

If you define multiple devices in a silo for Backup to use, you do not need to specify them in any particular order in the Jukeboxes resource. The only restriction is that the list of device names in the Devices attribute must be in the correct order with respect to the list in the STL Device Names attribute.

## Shared Devices

The basic elements of device sharing have been implemented in Backup. However, due to current limitations in the SCSI hardware on Backup platforms, none of the device-sharing functions have been implemented. If you issue device reservation commands there are no errors, but the devices are not reserved. The STL Device Sharing and STL Device Reservation attributes in the Backup administration program and `nsradmin` have no effect on the behavior of Backup.

# Device Cleaning

Do not enable automated cleaning for the silo in Backup. Refer to the documentation for your silo management software to find out how to clean devices in your silo. You cannot use the Backup automated device cleaning in a silo because it depends on fixed slot numbers.

APPENDIX **A**

# Backup Functionality

Backup client-server technology uses a network Remote Procedure Call (RPC) protocol to back up your data. The Backup server software consists of several daemons and programs that oversee the backup and recover processes, as well as storage management client configurations, a client file index, and a media database. The Backup client software includes the `nsrexecd` daemon and user interface programs.

This chapter provides a brief, simplified overview of how Backup performs a backup and recovery. Illustrations of the backup and recovery processes provide a graphical overview of the storage management process Backup uses.

## How Backup Backs Up Data

Backup calls upon several daemons and programs when a request for a backup is received. The daemons coordinate the tasks associated with a backup or recovery and record information about what was backed up and the media containing the backed-up data.

### Backup Daemons and Programs

This section provides a description of the Backup server and storage node daemons and programs, which contact the client for a backup and maintain the server's client file index and media databases. (For more information about storage nodes, see

"Backup Operations With Storage Nodes" on page 257.) The Backup for UNIX man pages contain further details about the Backup daemons and programs. The following table describes the server daemons and programs.

**TABLE A-1** Server Daemons and Programs

| Daemon/Program | Function |
|---|---|
| `ansrd` | This daemon monitors an active `save` or `recover` session; agent process spawned by `nsrd` in response to a `save` or `recover` session. |
| `asavegrp` | This daemon monitors the progress of individual save sets; agent process invoked by `savegrp`. |
| `nsrck` | This daemon checks the consistency of the client file index; invoked by `nsrd` whenever the consistency of the client file index needs to be confirmed. |
| `nsrd` | This daemon provides an RPC-based `save` and `recover` service to Backup clients; master Backup daemon. |
| `nsrim` | This daemon automatically manages the server's client file index; invoked at the end of a `savegrp`. |
| `nsrindexd` | This daemon provides a method for inserting entries in the client file index based on information passed by the `save` program. |
| `nsrmmd` | This daemon provides device support, generates mount requests, and *multiplexes* save set data during a multi-client backup (`nsrd` can start several `nsrmmd` daemons, up to the number specified in the device's Target Sessions attribute); media multiplexor daemon. |
| `nsrmmdbd` | This daemon provides media and save set database management services to the local `nsrd` and `nsrmmd` daemons and records entries in the media database; media management database daemon. |
| `savegrp` | This program runs a group of Backup clients through the `save` process. |

The `nsrd` master Backup server daemon is responsible for several tasks:

- Starting other daemons
- Allocating media daemons on server and storage node machines
- Authorizing backup and recover services for the client
- Contacting clients for scheduled backups
- Maintaining Backup configuration information
- Monitoring backup and recover sessions
- Maintaining server statistics and message logs

The nsrd Backup server daemon calls on the nsrexecd Backup client daemon and several client-side programs when it receives a scheduled or on-demand backup request. The ansrd temporary server agent daemon starts on the Backup server to monitor the progress of the backup session. The following table describes the client-side daemons and programs.

**TABLE A-2** Functions of Client Daemons and Programs

| Daemon/Program | Function |
|---|---|
| nsrexecd | This daemon authenticates the Backup server's remote execution request and executes the save and savefs commands on the client. |
| recover | This program browses the Backup server's client file index and restores the specified file to primary disk storage. |
| save | This program sends specified files in a multiplexed data stream to the Backup server for backup to media by nsrmmd and entry in the client file indexes and media database by nsrindexd. |
| savefs | This program sends information about the save sets to back up for the client; identifies save set data modified since the previous level save. |

# Events During a Scheduled Backup Backup

When you configure a *backup group* on the Backup server, you schedule a start time for the backup group. The nsrd server daemon starts the server's savegrp program for the backup group at the scheduled time.

The savegrp program queries the client resources configured on the Backup server to determine:

- Which clients configured on the server are members of the scheduled group
- What level of backup (save) to perform
- How many save sets to run concurrently, determined by the parallelism value set on the Backup server
- When the most recent backup of the group occurred

If any of this information is not available on the Backup server, savegrp sends a request to the nsrexecd client-side daemon to run savefs on each client assigned to the backup group to gather the necessary details.

The savefs program tells savegrp which objects to back up for the client. After savegrp receives information about the objects to back up, savegrp assembles a work list for the server. The work list specifies the order in which clients are contacted for backup. The order of the work list is determined by the Client Priority attribute in the Clients resource. The client with the lowest value in the Client Priority attribute is contacted first.

If problems were encountered with the client file index during the previous backup session, `nsrd` invokes the `nsrck` daemon to check the consistency and state of the Backup server's client file indexes and media database. Then, `nsrd` starts the `nsrindexd` client file index insertion daemon.

The `savegrp` program contacts the first client on the server's work list. The client's `nsrexecd` is invoked and starts a `save` session of the first save set listed on the server's work list. The `save` program passes to `nsrd` all save criteria, such as group, client, save sets, storage node affinity, and level of the save data. With this information, `nsrd` determines the pool of volumes that will store the data and forwards the information to the appropriate media daemon, on either the Backup server or on a storage node controlled by the Backup server.

The `nsrmmd` media daemon:

- Sends a message to the console of the Backup server, requesting a mount of the media assigned to the volume pool indicated by `nsrd`
- Writes the data sent by `save` to storage media
- Forwards storage information to `nsrmmdbd` for recording in the Backup server's media database

Any time there is a lull in save set activity from the client, the Backup server attempts to find another save set in the group to keep the process moving. The `savegrp` program attempts to concurrently back up as many save sets as possible, up to the limit set by the parallelism attribute in the Backup server's configuration, to use the backup devices to their maximum potential.

The `savegrp` program repeats the process for each item on the server's work list until all clients in the group are backed up. Before the `savegrp` is completed, if the Backup server is part of the group being backed up or the server is not part of any enabled group, `nsrim` is invoked and the Backup server's *bootstrap* file is backed up. When the bootstrap backup is completed, a bootstrap printout is sent to the default printer configured for the Backup server. Keep the bootstrap printout in a safe place in case you need to restore the Backup server.

The final results of the `savegrp` execution are sent back to the server and are included in the *savegroup completion report*, which is sent through email to *root*.

The following figure shows how all the Backup client and server daemons and programs interact during a scheduled save.

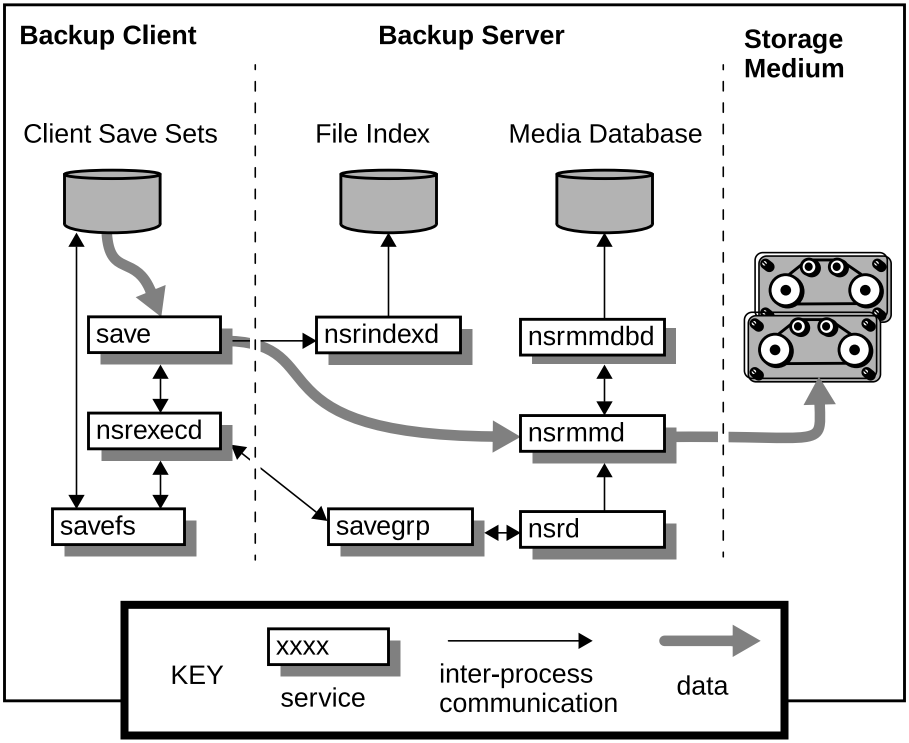


**FIGURE A-1** How Backup Daemon Processes and Programs Interact During a Scheduled `save`

# How Backup Recovers Data

When Backup receives a `recover` request from a client, the server's `nsrd` daemon contacts the server's `nsrmmd` media daemon. The `nsrmmd` daemon contacts the server's `nsrmmdbd` media database daemon to determine which media contain the save set requested by `recover`. After the save set's media location is obtained, `nsrmmd` issues a mount request, the media is positioned to the beginning of the save set, and the save set stored on the mounted media is passed to `nsrmmd`. The media daemon forwards the save set to the client's `recover` program, which restores the data to the client's filesystem.

When the server's nsrmmdbd media database daemon cannot locate the required volumes, or when there are no nsrmmd daemons (enabled devices) on the Backup storage node or server, the recover request fails. Backup displays the following message on the client machine that requested the recover:

```
NSR server client: no matching devices on server-or-storage-node
```

Backup also displays the following message in the Pending display for the Backup server:

```
media notice: no matching devices on server-or-storage-node for
recover by client client
media notice: enable or check device(s) on server-or-storage-node
```

The following figure shows how the Backup server and client daemons and programs interact while recovering data to a Backup client.

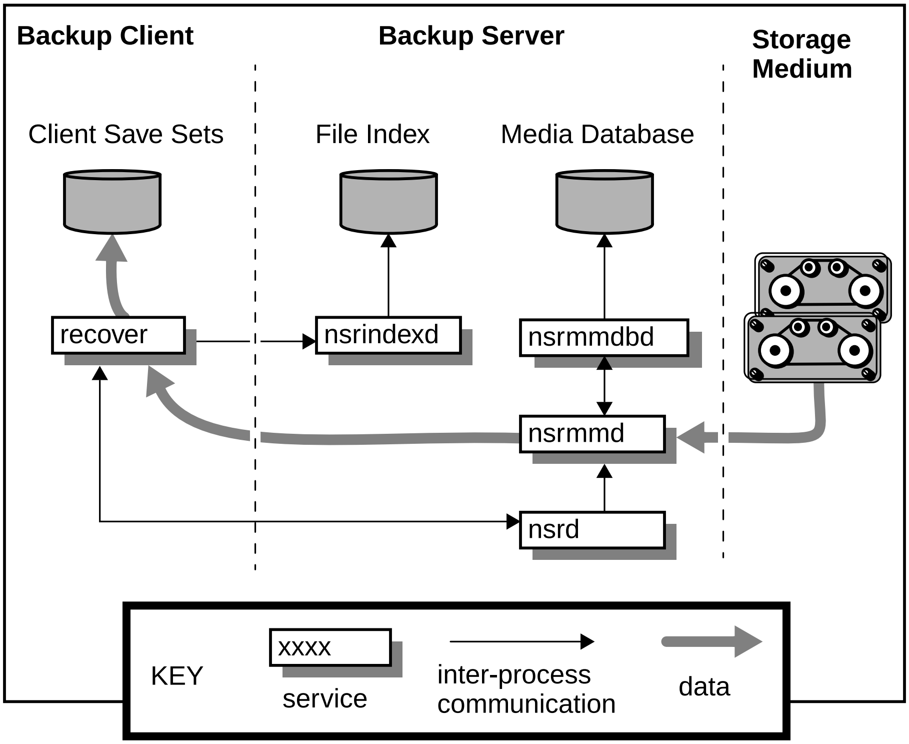


**FIGURE A-2** How Backup Daemon Processes and Programs Interact During a Recover Session

# Backup Operations With Storage Nodes

A storage node is a machine that contains Backup client and media management daemons and programs. A storage node is connected to one or more storage devices that are used in Backup operations, such as backup and recovery. Media management daemons on the storage node machine read and write data on storage volumes.

Backup and recover operations with storage nodes are very similar to backup and recover operations on the Backup server. The major difference is where the data resides.

A storage node is controlled by a Backup server. The Backup server's `nsrd` master daemon starts the `nsrmmd`, media management daemon, on the storage node, and during backup, the Backup server routes appropriate data to each media management daemon according to the Storage Node Affinity attribute in the Clients resource.

After data is written to the storage devices connected to storage node machines (remote devices), the Backup server tracks the location of the data. On the Backup server, the `nsrindexd` daemon writes entries in the client file indexes to track the location of each file in a save set, and the `nsrmmdbd` daemon writes entries in the media database to track the location of each save set on media.

The following figure shows how the Backup server, client, and storage node daemons and programs interact during a scheduled save in a Backup setup that includes a storage node and a remote storage device.

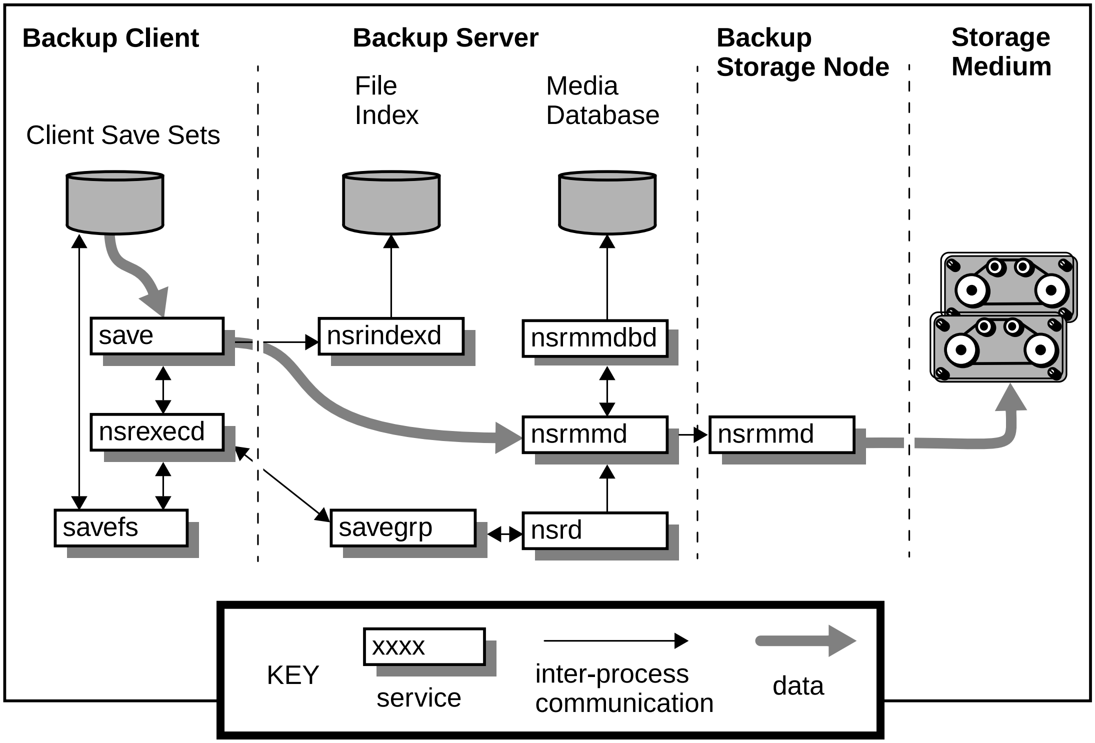


FIGURE A-3 How Backup Daemon Processes and Programs Interact During a Save Session With a Storage Node

When the Backup server receives a request from the client's `recover` program, it contacts `nsrmmd`, the media management daemon, on the storage node instead of on the server machine. The `nsrmmd` daemon reads the data from media and sends the data to the Backup client, where the `recover` program rebuilds the data.

The following figure shows an example of how a recover process works in a Backup setup that includes a storage node and a remote storage device.

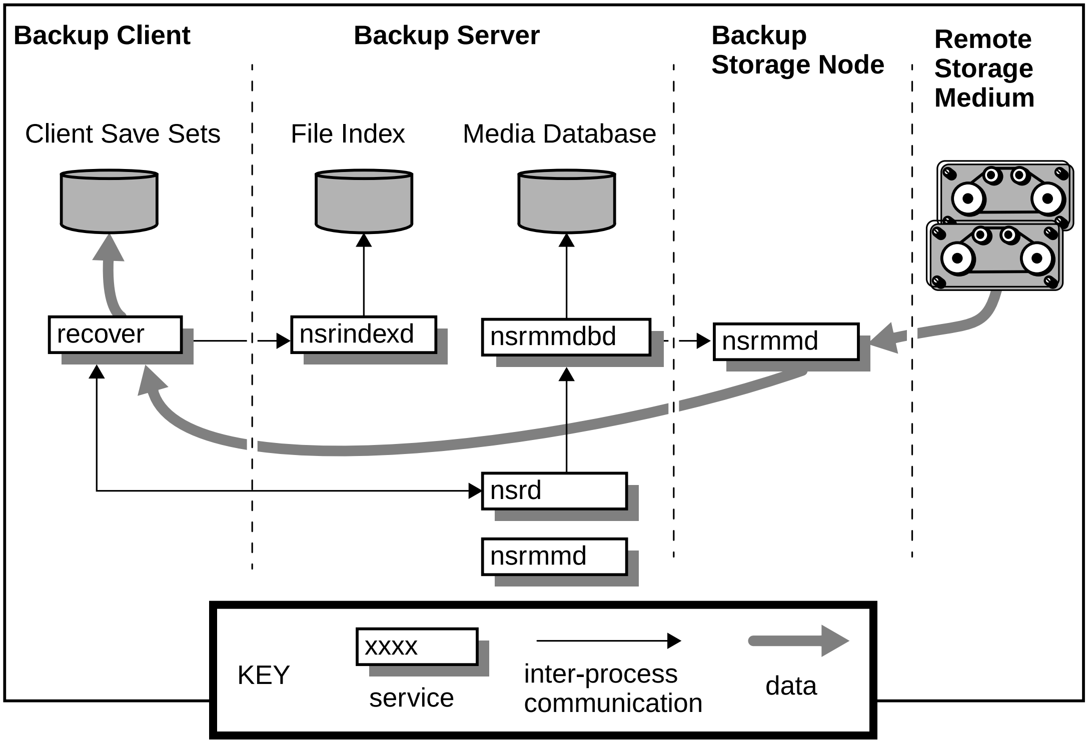


**FIGURE A-4** How Backup Daemon Processes and Programs Interact During a Recover Session With a Storage Node

APPENDIX **B**

# Command Line Reference Utilities

The Backup software includes a graphical user interface (GUI) as well as a command line interface. For instructions on how to use the GUI, see the online help included in the program, which you start by invoking the `nwadmin` command at the shell prompt.

This appendix provides an abbreviated reference for some of the options available through the command line interface. The information is organized by the tasks to which they relate. The online manual (man) pages, included with your Backup software, are available for more detailed information and examples about each command.

To view a man page, make sure that the MANPATH environment variable includes the path where you installed the Backup man pages, then enter `man` *command-name*, for example, `man nsrjb`. To display a man page that explains the man pages, enter `man man`. To print a copy of the entire collection of Backup man pages, enter the `troff` command at the shell prompt with the options shown in this example:

```
% troff -t -man ‘nsr_man -l’| lpr -t -P printer-name
```

The command for your machine can vary (for example, your print command may be `lp` instead of `lpr`), depending on the operating system and the version of PostScript™ or troff software you have installed.

## Enabler Entry and Registration

When you enter an enabler code, you unlock features of Backup that you can use for 45 days. To continue to use Backup after the 45 days expire, you must follow the instructions provided in “How to Register and Authorize Your Software” on page 13 and register your enabled software. When you register your software, a unique authorization code is generated that is keyed to your specific system information and enabler code. After you receive and enter the authorization code, you can use the Backup software indefinitely.

## nsr_ize

The interactive nsr_ize program installs or removes Backup software and files to or from a computer with the one of the following operation systems: AIX, IRIX, SCO, SunOS, or UnixWare. Informational prompts guide you through a series of questions, many of which already provide default answers to use for a standard environment.

The nsr_ize program modifies several system administration files, including /etc/rpc. If you use YP, modify the YP master's /etc/rpc file with the same modifications that nsr_ize makes to the local copy of /etc/rpc.

The following example describes the format and options available for the nsr_ize program:

```
nsr_ize [-i | -r -u] [-c | -s] [-kmnqxv]
```

- Use the -c option to tell nsr_ize to install or remove only the client software.
- Use the -i option to install the Backup software and associated files.
- Use the -k option to kill the Backup daemons without confirmation.
- Use the -m option to tell nsr_ize not to install or remove the Backup man pages.
- Use the -n option to tell nsr_ize not to perform actions that change the file system. When you use the -n option, nsr_ize prints the installation script without performing the commands.
- Use the -q option to run nsr_ize in quiet mode.
- Use the -r option to remove the Backup software and associated files.
- Use the -s option to tell nsr_ize to install or remove only the server software.
- Use the -u option to prepare your system for a Backup software upgrade. The existing Backup software is removed, but the nsr.res file, client file indexes, server bootstrap, and media database are preserved.
- Use the -v option to run nsr_ize in verbose mode.
- Use the -x option to set the debug flag.

# nsrlic

The nsrlic program generates reports about all the license information currently active on the Backup server. This command queries the Backup resource database, and formats and displays the results to standard output. You do not need to be root Administrator to invoke nsrlic.

If you enter nsrlic at the shell prompt without optional flags, you receive a report, similar to the following example, for the server that you invoked the command from:

```
SERVER (UNIVERSAL) CLIENT LICENSES
                     Available: 10
                          Used: 0
          Borrowed from Server: 0
                     Remaining: 10
             Connected Clients: ;
               Defined Clients: ;
WORKSTATION CLIENT LICENSES
                     Available: 0
                          Used: 0
                     Remaining: 0
             Connected Clients: ;
               Defined Clients: ;
SERVER CLIENT TYPES
                           AIX: 0
                            HP: 0
                       Solaris: 0
                         SunOS: 0
             Windows NT Server: 0
                       NetWare: 0
WORKSTATION CLIENT TYPES
                           DOS: 0
                     Macintosh: 0
                          OS/2: 0
                  Windows 3.1x: 0
                    Windows 95: 0
        Windows NT Workstation: 0
                        Others: 4
```

The following example describes the format and options available for the nsrlic program:

```
# nsrlic -vi -s server
```

- Use the `-i` option to use `nsrlic` in the interactive mode. In this mode, you can request different reports, refresh the information, or switch to a different server. The interactive mode provides a prompt and displays the choices available:

```
connecting to jupiter...
Available commands are:
   summary - display a summary report of licenses
   detail - display a detailed report of licenses
   connect [server name] - connect to server
   help - list command helps.
   quit - quit out of nsrlic command.
nsrlic>
```

  The information is requested once and cached until you issue another connect command at the `nsrlic` prompt.

- Use the `-s server` option to select a specific Backup server to query. If you omit this option, the server from which you invoked the `nsrlic` program is queried.
- Use the `-v` option to generate a more detailed, verbose report. In addition to the number of licenses or the number of clients, a list of connected and defined clients is gathered and displayed.

## NSR license

The NSR `license` resource describes each Backup software feature that you entered an enabler code for, as well as the permanent authorization code, once entered. To inspect the `NSR license` resource on your Backup server, become root Administrator and use the GUI to view the Registration window or enter the following command at the shell prompt:

```
# nsradmin -c “type:NSR license”nsradmin -s server-name
# nsradmin print type:NSR license
```

You can create, enable, or authorize a `NSR license` resource from within the GUI; however, you must use the `nsrcap` command to update an existing `NSR license` resource.

# nsrcap

The nsrcap program enters a unique enabler code into the Backup server's nsr_license resource that enables you to use features in the Backup software that you installed. You can use the nsrcap program to enter the enabler code for a new feature, or you can use the nsrcap program to enter an enabler code that upgrades or downgrades Backup software features that you are already using.

The following example describes the format and options available for the nsrcap program:

```
# nsrcap [-vn] {-c | -u | -d} enabler-code
```

To use the nsrcap program, you must log in as Administrator and become root on the Backup server and specify only one of the following command options:

- Use the -c option to enter an enabler code that enables you to use a feature that is not already installed. You can only load a feature once; an error is returned if you try to load the enabler more than once.
- Use the -d option to enter an enabler that downgrades an existing Base or Jukebox enabler. After you downgrade the enabler, you cannot return to the previous level enabled on your system. *Do not* use the -d option unless instructed to do so by Sun Technical Support.
- Use the -u option to enter an enabler that upgrades an existing Base enabler. (The -u option only works for the server enabler code). After you upgrade the enabler, you cannot return to the previous level enabled on your system.

The nsrcap program has two additional options that you can elect to use when you enter the one of the following command options:

- Use the -v option if you want the nsrcap program to display verbose information that describes the enabler entered.
- Use the -n option if you want to inspect the enabler code for validity. When you specify the -n option, the enabler code you enter on the command line is inspected, but is not entered into the Backup server's nsr_license resource.

# nsr_shutdown

The nsr_shutdown command identifies and kills the Backup processes on a Backup server. Use the command whenever you need to install or remove Backup software. You must become root on the system to use the nsr_shutdown command.

The following example describes the format and command options available for nsr_shutdown:

- Use the `-a` option to kill all of the Backup daemons. The option has the same effect as the `-A`, `-d`, and `-s` options combined.
- Use the `-A` option to kill any `nsralist` processes.
- Use the `-d` option to kill the Backup server daemons. If you do not specify any options, the `-d` option is assumed by default.
- Use the `-n` option to echo the kill command without a real shutdown.
- Use the `-q` option to perform a quiet shutdown, without prompts for confirmation.
- Use the `-s` option to kill any `savegrp` and `nsrexecd` processes.
- Use the `-v` option to echo commands and their arguments as `nsr_shutdown` executes them.

## User Interface Startup

You can use the Backup software through a command line interface or a GUI. You can start the administrative programs from any machine on the network; however, only users with administrator privileges can make changes. You can use the user programs for backup and recovery, as well as the optional archive and retrieve features on any client that has the feature enabled in the client resource.

For server selection, the client commands are classified into two groups: administration and operation. The administration commands include `nwadmin`, `nsrwatch`, and `mminfo`. The operation commands include `save`, `savefs`, and `recover`. Both groups of commands accept a `-s server` option to explicitly specify a Backup server.

When a server is not explicitly specified, the operation commands use the following steps to locate one. The first available server found is the one used.

1. The system where the current directory is actually located is determined. This is either an NFS server or the local system. If that system is a client of a Backup server as determined by a RAP query, then that Backup server is used. If more than one server backs up the current directory, one server is chosen and an informational message is printed showing the other server's names.

2. The system where the current directory is actually located is examined to see if it is a Backup server. If it is, then that system is used.

3. The local system is examined to see of it is a Backup server. If it is, then the local system is used.

4. If a Backup server is still not found, then the system with the hostname `nsrhost` is used.

# nsradmin

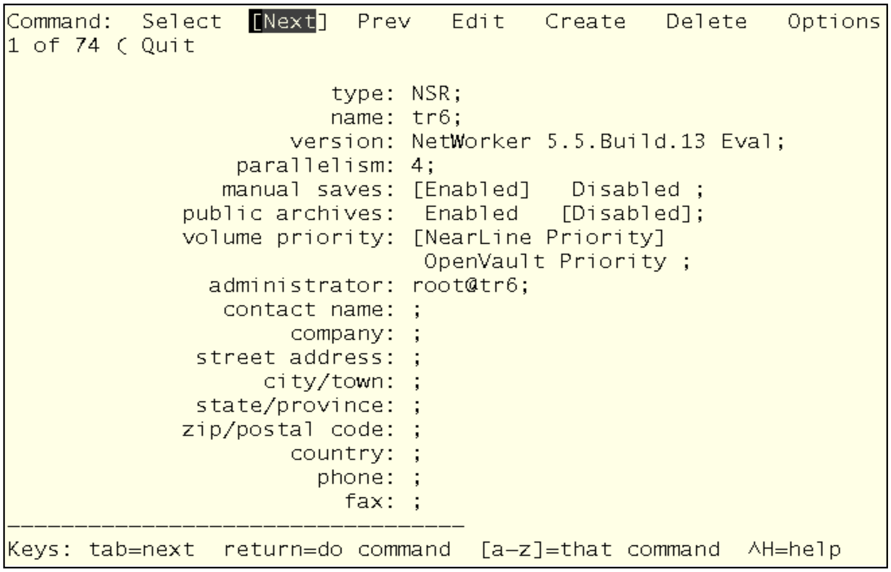
The nsradmin program is an administrative program for the Backup system that uses the command line. Typically, nsradmin monitors and modifies Backup resources over the network. Commands are entered on standard input, and output is produced on standard output.

If you enter the nsradmin command without command options, the program opens with a command prompt for you to enter additional options as needed:

```
nsradmin>
```

If you enter help at the nsradmin prompt, help text explaining the nsradmin commands and their usage displays.

If you enter visual at the nsradmin prompt, a cursor based version of the interface displays, as shown in the following figure. Use the Tab or directional arrow keys to toggle between the menu choices. Press Return to select a choice. To view the online help available for nsradmin, press the Control and H keys simultaneously.



```
Command:  Select  [Next]  Prev   Edit   Create   Delete   Options
1 of 74 ( Quit

                       type: NSR;
                       name: tr6;
                    version: NetWorker 5.5.Build.13 Eval;
                parallelism: 4;
               manual saves: [Enabled]   Disabled ;
            public archives:  Enabled   [Disabled];
            volume priority: [NearLine Priority]
                              OpenVault Priority ;
              administrator: root@tr6;
               contact name: ;
                    company: ;
             street address: ;
                  city/town: ;
             state/province: ;
            zip/postal code: ;
                    country: ;
                      phone: ;
                        fax: ;
----------------------------------
Keys: tab=next  return=do command  [a-z]=that command  ^H=help
```

**FIGURE B-1** Visual Mode for nsradmin Program

Note that the value for Version may differ from the example shown above, depending on the final build of the software and whether you install patches for that version at a later date.

The following example describes the format and command options available for the `nsradmin` program:

```
nsradmin [-c] [-i file] [-s server] [-p prognum] [v version] [query]
nsradmin [-c] [-i file] [-f resource-file]
[-t typefile] [query]
nsradmin [-i file] [-s server]
```

- Enter the `-f resource-file` option to use the Backup resource file you specify for `resource-file` instead of opening a network connection. Do not use this option if the Backup server is currently running a backup. You can use multiple `-f` and *resource-file* arguments to start `nsradmin` with access to more than one file at a time.
- Enter the `-i file` option to tell Backup to take input commands from a file instead of from standard input. The interactive prompt is not printed when you use the `nsradmin` program in this mode.
- Enter the `-p` *program* option to use the given RPC program number instead of the standard program number. The standard program number is 390109. Generally, you should use this option only to debug problems that you encounter.
- Enter the `-s server` option to open a connection to a specific Backup server. This command is useful when you want to limit the number of resources polled if there are many servers, or to administer Backup when the RAP location service is not working.
- Enter the `-t typefile` option to use the alternate file `typefile` to define RAP types.
- Enter the `-v` *version* option to bind to the Backup RAP service with the given version number. The default value for version is 2. Generally, you should use this option only to debug problems that you encounter.
- Specify the *query* option, in the form of an attribute list, to perform an edit operation:

```
attribute ::= name [: value [, value]*]
```

An attribute is a name optionally followed by a colon, followed by zero or more values, with values separated by commas. A comma at the end of a line continues the line.

```
attribute list ::= attribute [; attribute]*
```

An attribute list is one or more attributes separated by semicolons. A semicolon at the end of a line continues the line. The list is ended by a newline character that is not preceded by a comma or semicolon.

```
name: mars;
type: NSR client;
remote access: mars, venus, jupiter
```

At each `nsradmin` input prompt, you enter a command name and optional arguments. You can shorten command names to the smallest unique string, for example, you can enter `p` for the `print` command. You specify command arguments in the form of an attribute list. Most `nsradmin` commands operate on a set of resources returned by a query. The query is specified as an attribute list that is used to match resources with the following rules:

- The resource must match all the given attributes.
- If more than one value is specified, the resource can match any one of the values.
- The values in a query may be in the form of regular expressions. A pattern match is attempted against all resources that contain the specified attribute.
- If an attribute is specified with no value, the resource must contain an attribute of that name.

If the query has only one name and no values, the `nsradmin` program tries to determine the query based on the name. If the name is a hostname, the query is made for all the resources on the given host. Otherwise, the name is interpreted as a type name, and all resources of that given type are selected.

## Command Options for `nsradmin`

The following list describes the commands available and their function:

- **`bind`** *query*

  To bind to the service that owns the resource described by query. If a query is not specified, send the queries to the RAP Resource Directory, and update, create, and delete commands to the service that owns the resource being changed. On failure, the previous service continues to be used.

- **`create`** *attribute-list*

  To create a resource with the given attributes.

- **`delete`** *query*

  To delete the resources that match the current query. If a query is specified, it becomes the current query.

- **`edit`** *query*

To edit the resources that match the current query. If a query is specified, it becomes the current query. When the editor exits, `nsradmin` applies update, delete, and create operations based on the changes to the resources. Do not edit the resource identifier attribute, but do write out the file before you exit the editor.

- **`help`** *command-name*

**`?`** *command-name*

To print a message describing a command. If no command name is given, a synopsis of all the commands is printed.

- **`print`** *query*

To print the resources that match the current query. If a query is specified, it becomes the current query. If the current show list is not empty, only the attributes named in the show list are displayed.

- **`server`** *server-name*

To bind to the given Backup server name. If no server is specified, the RAP location service is used. On failure, the previous server continues to be used.

- **`show`** *name*

To add names to the show list if a name list (really an attribute list with no values) is specified. Only these attributes are displayed in subsequent print commands. If no name list is given the show list is cleared, resulting in all attributes being shown.

- **`types`**

To print a list of all known types.

- **`update`** *attributes*

To update the resources given by the current query to match attributes.

- **`quit`**

To exit the `nsradmin` program.

- **`option dynamic:`***choice***`;hidden:`***choice***`;resource id:`***choice*

To enable some options to change the display of resources. With no arguments it displays the current options; with a list of options it turns the specified ones on. The option command sets the given display options. Options are separated by semicolons, and you can give them an explicit value of either *on* or *off*.

The valid options are:

- dynamic, which causes `nsradmin` to display all dynamic attributes, even the normally hidden ones.

  - hidden, which causes `nsradmin` to display all attributes, even the normally hidden ones.
  - resource id, which causes `nsradmin` to display the resource identifier of each resource. The resource ID is a number that Backup uses internally to provide sequencing and uniqueness.
- `unset` **`dynamic;hidden;resource id`**

  To turn off the specified option.
- `.` *query*

  To set the current query, if a query is specified, without printing the results of the query. Otherwise, the current query, show list, server binding, and options are displayed.

## Resources Available in the `nsradmin` Program

The `nsradmin` program provides a character-based interface to manage the same resources available through the `nwadmin` program. These include:

## NSR Client

The **NSR client** resource describes the files that are saved, the backup schedule, the directive used to omit files from the save, the length of time the files' index entries should be kept in the on-line file and media indexes, the users given access to back up, browse, and recover a client's files. To edit the NSR client resources for a Backup server use `nsradmin` or use the Backup Administrator GUI `nwadmin`.

The **NSR client** resource has the following attributes:

- The **name** attribute specifies the hostname of a Backup client.
- The **server** attribute specifies the hostname of a client's Backup server.
- The **archive services** attribute specifies if a system can use archive services. To use this attribute archive support must be enabled on the server first.
- The **schedule** attribute specifies the name of the schedule controlling the backup levels for the save sets listed in the **save set** attribute.
- The **browse policy** attribute specifies the name of the policy controlling the length of time entries will remain in a client's on-line file index.
- The **retention policy** attribute specifies the name of the policy controlling the length of time entries will remain in the media index before they are marked as recyclable.
- The **directive** attribute specifies the directive used for backing up a client.
- The **group** attribute specifies the group a client is a member of. The group controls when scheduled backups are performed on the client.

- The **save set** attribute lists the path names to be saved for a client. When a client requires different file systems to be saved on different schedules, a client resource is required for each file system and schedule.
- The **priority** attribute specifies the backup priority given to a client where priority 1 is the highest, 1000 is the lowest. Automated savegroup's will attempt to back up clients with higher priorities before clients with lower priorities.
- The **remote access** attribute specifies a users access to back up, browse, and recover a client's files. Additional users, hosts, and netgroups may be granted permission to access a client's files by adding their names to this attribute. Netgroup names must be preceded by an ampersand (&). Input of the form `user@host` or `host/use`r, grants access to a client's files to the specified users.
- The **remote user** attribute:
  - Specifies the user login name a Backup server will use to authenticate itself with a client, who has accessed the network through `rsh` or `nsrexecd`.
  - Allows the Backup server (when run with the `savegrp -p` command) to determine which files to save.
  - Allows certain clients, (such as NetWare fileservers) to gain access to files being backed up. This procedure only works when the `remote user` attribute is used along with the `password` attribute.
- The **password** attribute is used by `savegrp` to initiate the commands `savefs` and `save` on a client machine. The commands `savefs` and `save` use the password to gain access to files being backed up. If a password is given, then the `remote user` attribute for the client resource must also be defined.
- The **backup command** performs a remote backup of client's data and save sets. This command can also perform pre and post backup processes. The prefix of the specified value must begin with `nsr` or `save`.
- The **executable path** attribute specifies the path used by the Backup server for executing commands on the client.
- The **server network interface** attribute specifies the network interface the server uses for saves and recovers.
- The **aliases** attribute specifies the aliases for a client machine that queries can match.
- The **owner notification** attribute sends the contents of status messages to the owner/primary user of a system.
- The **statistics** attribute consists of: the size of the client's online file index, the number of kilobytes used and the number of entries in the index.
- The **index save set** attribute specifies save set, residing in a client's file index, to purge when an index operation is set to purging oldest cycle.
- The **index message** attribute is the status message resulting from the previous index operation.
- The **index operation start** attribute indicates the starting time of the current index operation. This attribute is a null string ("") when the operation is Idle.
- The **index progress** attribute indicates the progress an index has made towards finishing the current task. This attribute is blank when the operation is Idle, and is expressed as a percentage.
- The **index operation** attribute specifies the current index operation.

- The **parallelism** attribute indicates the maximum number of saves that should be run simultaneously on a single client.
- The **archive users** attribute specifies the users given access to the archive services on a client. This attribute can only be set if archive support has been enabled on the server.
- The **application information** attribute specifies a client's application information.
- The **storage nodes** attribute specifies the storage nodes available to a client for saving data. A client's saves are directed to the first storage node that has an enabled device and a functional media service.
- The **clone storage nodes** attribute specifies the storage nodes available to a storage node whose data is being cloned. Cloned data originating from a storage node will be directed to the first storage node that has an enabled device and a functional media service.

The following is an example of a **NSR client** resource used to define a client, called `saturn`, backing up all of its files to the Backup server mars:

```
type: NSR client;
name: saturn;
server: mars;
archive services: Disabled;
schedule: Default;
browse policy: Month;
retention policy: Quarter;
directive: ;
group: engineering;
save set: h:\, c: \usr,  c:\usrsrc;
remote access: venus, sam@*, jupiter/john;
remote user: operator;
password: ;
backup command: ;
aliases: saturn.legato.com;
archive users: ;
storage nodes: nsrserverhost;
clone storage nodes: ;
```

## NSR Device

The **NSR device** resource describes each storage device used by a Backup server. To edit the **NSR device** resources for a Backup server use `nsradmin` or use the Backup Administrator GUI (`nwadmin`).

The **NSR device** resource has the following attributes:

- The **name** attribute specifies the path name for a device. For systems that optionally support "Berkeley style" tape positioning on close, the BSD style tape device name should be used. For optical disks the path name is generally the "c" partition.

  To facilitate interaction with external media management services, a logical device type has been defined. When interacting with such services, the device into which a volume is loaded may be determined by the media management service. A logical device is used to define a Backup device resource.

  At the time of definition the name of a device is not related to any specific device. The default for both the media type and family are set to **logical.** The name, type, and family are not determined until the media management service has loaded a volume into a device in response to a request made by Backup. The name, type, and family of the actual device are then stored in the attributes **logical name, logical type,** and **logical family,** respectively. The association between the logical device and the actual device last only as long as a volume is loaded into the device and allocated for use by Backup.
- The **media type** attribute specifies the media type used by a device. Some of the possible values for this attribute are:
  - **4mm**, 4mm digital audio tape (1 Gbyte)
  - **8mm**, 8mm video tape (2 Gbyte)
  - **dlt**, digital linear tape cartridge (10 Gbyte)
  - **vhs,** VHS data grade video tape (14 Gbyte); **3480** – high-speed cartridge tape (200 Mbyte)
  - **logical**, used when interacting with an external media management service.
- The **enabled** attribute indicates whether a device is available for use.
- The **read only** attribute indicates whether a device is reserved for read only operations, such as recover or retrieve.
- The **target sessions** attribute specifies the target number of saves for a device, and used for load-balancing. Once all the devices have reached their corresponding target number, additional sessions are allocated equally across all devices.
- The **media family** attribute specifies the class of storage media, as determined from the media type:
  - **tape**, tape storage device
  - **disk**, disk storage device
  - **logical**, external media device.
- The **message** attribute specifies the last message from a Backup server regarding a device, such as the progress or rate of an operation.
- The **volume name** attribute is monitors the mounting and unmounting of volumes for a device.
- The **write enabled** attribute indicates if writing to the current volume is allowed.
- The **volume operation** attribute manipulates media volumes currently in the device, through several operations:

  - The **Unmount** operation releases the device.
  - The **Mount** operation mounts the loaded volume onto the device.
  - The **Verify label** operation reads the volume's label, volume's attributes and sets the volume expiration.
  - The **Verify write** `time` operation sets the volume write time attribute.
  - The **Label or Label** `without mount` operations create new labels for volumes.
  - The **Eject** operation ejects volumes from the device.
  - The **Monitor device** operation periodically checks the device to determine whether a volume has been loaded into the device. When a volume containing a readable Backup label is loaded, the volume is listed in the Backup server's media database, and the volume is writable the volume is mounted with write permissions. Otherwise the volume is mounted read only.
- The **volume label** attribute is set by the **Verify label** operation and may be an input to the **Label** operation.
- The **volume default capacity** attribute is used by the **Label** operation if the **volume current capacity** attribute is blank. This attribute enables the override of default sizes when using devices (or tapes) with different capacities than the defaults.
- The **volume current capacity** attribute determines the capacity of a volume during the **Label** operation.
- The **volume expiration** attribute specifies a volumes expiration date, which is set by the **Verify label** operation.
- The **volume pool** attribute specifies the pool a volume belongs, or has been assigned to.
- The **NSR operation** attribute specifies the current operation being performed by the device.
- The **minor mode** attribute reports the current status of a device.
- The **statistics** attribute reports the statistics on the operation of a device. The statistics include:
  - **elapsed**, the time of operation
  - **errors**, the number of errors
  - **last rate**, the last writing rate
  - **max clients**, the maximum number of concurrent clients
  - **file marks**, the number of file marks written
  - **rewinds**, the number of rewinds
  - **files skipped**, the number of files skipped
  - **records skipped**, the number of records skipped
  - **current file**, the current file number
  - **current record**, the current record number
  - **seek files**, the relative number of files being spaced over

    - **seek records**, the relative number of records being spaced over
    - **estimated kb**, the total estimated amount read/written on a volume
    - **amount kb**, the total amount read/written on the volume, in kb
    - **file amount kb**, the current amount read/written on this file, in kb
    - **sessions**, the current number of sessions assigned to this device
- The **cleaning required** attribute indicates whether a device needs to be cleaned. If the value of this attribute changes from yes to no and the value of **date last cleaned** is not updated then **date last cleaned** is set to the current time. Backup will set this attribute to yes if the device is scheduled to be cleaned. Then the notification **device cleaning required** is sent, indicating that a device needs to be cleaned.
- The **cleaning interval** attribute specifies the amount of time from **date last cleaned** until the next scheduled cleaning for a device.
- The **date last cleaned** attribute records the time and day a device was last cleaned.
- The **volume block size** attribute specifies the block size of a currently mounted volume.
- The **volumeid** attribute specifies the volume ID for a currently mounted volume.
- The **access count** attribute indicates the number of operations performed on a device since it's configuration as a Backup device.
- The **access weight** attribute indicates the weight of a single operation performed on a device. Each time a device is used its weight is increased and the less often the device will be selected for new operations.
- The **consecutive errors** attribute specifies the current number of consecutive errors resident on a device.
- The **max consecutive errors** attribute indicates the maximum number of consecutive errors allowed before the device will be disabled.
- The **operation arg** attribute specifies extra parameters about a device operation. Parameters are packed into a string and parsed.
- The **volume message** attribute indicates the result of the volume's last operation.
- The **volume write time** attribute indicates the time a save set was first written onto the volume.
- The **volume flags** attribute indicates new flags for the volume to operated on, during a "Label" or "Label without mount" operation.
- The **jukebox device** attribute indicates if a media device is in a jukebox
- The **unlabeled volume loaded** attribute indicates whether a volume loaded into a device has a readable Backup volume label.
- The **auto media management** attribute indicates whether automated media management for a device is enabled. If the value is set to **yes** then recyclable volumes loaded into the device may automatically be re-labeled by Backup for re-use and unlabeled volumes loaded into the device may be automatically labeled. A volume is considered to be unlabeled if the volume does not contain a label that may be read by this device. Volumes are considered unlabeled:
    - If a volume contains a label written at a density that can not be read by this device.

  - If a volume contains data written by an application other than Backup and does not have a label recognizable by Backup.
- The **logical name** attribute specifies the name for a logical device.
- The **logical type** attribute specifies the type for a logical device.
- The **logical family** attribute is the family associated with a logical device.
- The **connection process id** attribute specifies the process identifier that maintains the connection between external media management services and a mounted volume.
- The **connection message** attribute specifies error messages reported by a process connected to an external media management service.
- The **connection status** attribute specifies the exit status reported by a process connected to an external media management service.
- The **save mount timeout** attribute indicates the timeout value from an initial save mount request for a storage node, on which a device resides. If a request is not satisfied, the storage node will be locked from receiving save assignments, for "save lockout" minutes.
- The **save lockout** attribute indicates the amount of time a storage node will be locked from receiving save assignments.

The following is an example of a **NSR device** resource:

```
type:NSR device;
name:/dev/nrst8;
message:writing, done
volume name:mars.017;
media family:tape;
media type:8mm 5GB;
enabled:Yes;
write enabled:Yes;
read only:No;
target sessions:4;
volume label:mars.017;
volume default capacity:;
volume current capacity:5000 MB;
volume expiration:“Thu Sep 21 17:23:37 1996";
volume pool:Default;
volume flags:;
volume operation:;
volume write time:;
volume block size:32 KB;
volume id:32449;
accesses:199;
access weight:1;
consecutive errors:0;
max consecutive errors:20;
operation arg:;
volume message:;
NSR operation:;
minor mode:idle;
jukebox device:Yes;
statistics:elapsed = 257572, errors = 0, last rate = 397,
max clients = 3, file marks = 22, rewinds = 4,
files skipped = 1976, records skipped = 0,
current file = 2389, current record = 162,
seek files = 0, seek records = 0,
estimated kb = 0, amount kb = 6273,
file amount kb = 6273, sessions = 1;
cleaning required:No;
cleaning interval:2 weeks;
date last cleaned:“Tue Apr 11 15:10:32 1995";
auto media management:No;
unlabeled volume loaded:No;
logical name:;
logical type:;
logical family:;
connection process id:;
connection message:;
connection status:;
save mount timeout:30;
save lockout:0;
```

## NSR Directive

The **NSR directive** resource controls the files that are saved and the special handling specifications placed on certain file types. To edit the **NSR directive** resources for a Backup server use `nsradmin` or use the Backup Administrator GUI (`nwadmin`).

The **NSR directive** resource has the following attributes:

- The **name** attribute specifies the name of a directive resource. Names are displayed as choices when creating or updating Backup client resources.
- The **directive** attribute indicates the rules that define a directive.

The following is an example of a `NSR directive` resource, named “NTdirective”

```
type:NSR directive;
name:NT directive;
directive:"
 << / >>
 +skip : core
 skip : tmp
 << c:\usr\spool\mail >>
 mailasm : *
 << c:\nsr >>
 allow
";
```

## NSR Group

The `NSR group` resource controls when a group of Backup clients begin saving data and whether scheduled backups are started automatically each day. To edit the **NSR group** resources for a Backup server use `nsradmin` or use the Backup Administrator GUI (`nwadmin`).

The **NSR group** resource has the following attributes:

- The **name** attribute specifies the name of a group defined by the resource. The **name** is an option within the **NSR client** and **NSRpool** resources.

- The **autostart** attribute determines if a group will be saved automatically on a daily basis. The following operations can be invoked by **autostart**:
  - The **Enabled** operation starts saving group members data at the time specified in the **start time** attribute.
  - The **Disabled** operation disables the automatic save process specified for members of a group.
  - The **Start now** operation saves group members data immediately.
- The **autorestart** attribute controls whether a group is automatically restarted after an incomplete save.
- The **stop now** attribute aborts a groups save processes immediately.
- The **start time** attribute specifies the time of day when a group will begin a save.
- The **last start** attribute is the last time a group began a save.
- The **interval** attribute specifies how often a group runs an automatic save.
- The **force incremental** attribute forces an incremental backup of a savegroup, for an **interval** attribute less than 24 hours.
- The **client retries** attribute indicates the number of times failed clients should be retried before **savegroup** declares them failed. A client's save sets are retried by **savegroup** whenever savegroup would otherwise not be able to start a new save set.
- The **clones** attribute causes saves of a group to automatically make a clone for every save set backed up.
- The **clone pool** attribute specifies the pool where save set clones are sent.
- The **options** attribute specifies the options indicated for a group's save.
- The **level** attribute indicates the level a savegroup will use when started automatically by Backup. When **level** is not specified, the NSR Schedule for each client filesystem will be used to determine the level.
- The **printer** attribute specifies the printer to which bootstrap save set information will be printed to.
- The **schedule** attribute specifies the level of save that will be performed.
- The **schedule time** attribute specifies the time a save will be performed.
- The **inactivity timeout** attribute is the time a **savegroup** command waits for any kind of activity from the server before concluding that a **savegroup** descendant is hung.
- The **work list** attribute indicates the saves still not completed. The worklist indicates; the client name, the level of save, and the path to save.
- The **completion** attribute indicates the status of each save set that has been completed.
- The **status** attribute indicates the current status of a NSR group:
  - **idle**, indicates the group is inactive
  - **running**, indicates the backups are in progress
  - **cloning**, indicates backups are complete and clones are being made.

The following is an example of a **nsr_group** resource:

```
    type:NSR group;
    name:Default;
    autostart:Enabled;
    start time:“3:33”;
    options:Restartable;
    printer:lp2;
    inactivity timeout:30;
    work list:mars, incr, /g, mars, incr, index,
    completion:mars, /, succeeded,
"mars: / level=incr,     31 KB 00:01:01     72 files
```

## NSR Jukebox

The **NSR jukebox** resource describes the physical characteristics of each autochanger known to Backup by a single resource of type **NSR Jukebox**. To edit the **NSR jukebox** resources for a Backup server use `nsradmin` or use the Backup Administrator GUI (`nwadmin`).

The **NSR jukebox** resource has the following attributes:

- The **name** attribute specifies the name of a jukebox.
- The **model** attribute specifies the jukebox model.
- The **physical slots** attribute specifies the first and last physical slot numbers in the jukebox. The first slot number must be less than or equal to the last slot number, and specified as two separate attribute values. For Silo Tape Libraries this attribute is equal to the number of volumes allocated to a Backup server**.**
- The **control port** attribute specifies the path of the control port, for the jukebox robotics. Control commands are then sent to the jukebox, from the control port. For Silo Tape Libraries this attribute specifies the hostname and type of the Silo Tape Libraries server.
- The **devices** attribute identifies device pathnames for each device residing within a jukebox. The entries are listed in the same order as they were physically installed in the jukebox.
- The **number devices** attribute identifies the number of configured devices in the jukebox.
- The **write enabled** attribute indicates if the mounted volume can be written to.
- The **bar code reader** attribute indicates if Backup is using the bar code label from the media when a jukebox has a bar code label reader.
- The **match bar code labels** attribute indicates if Backup is using the bar code label, instead of a label template, when labeling media volumes.
- The **volume expiration** attribute specifies the expiration time for a volume that is currently being labeled, or specifies the time a volume within a jukebox will end interaction with external media management services.

- The **available slots** attribute specifies the slots containing volumes available to be written to by Backup requests. The slots are specified by a range which may be a single slot number or a pair of slot numbers separated by a dash. The first number is less than or equal to the second. When satisfying requests to mount a particular volume or slot, all of the volumes within the *physical slots* can be used.
- The **enabler code** attribute identifies the enabler code for a **NSR license** resource corresponding to a jukebox resource.
- The **operation** attribute identifies the current jukebox operation.
- The **operation message** attribute displays error messages when an jukebox operation fails.
- The **operation device** attribute passes device names to current operations.
- The **operation slots** attribute passes slots to current operations.
- The **operation options** attribute passes the mode of a volume to the current operation.
- The **operation barcodes** attribute passes volume tags or barcodes to the current operation.
- The **operation response** attribute identifies the default response to questions asked while performing an operation.
- The **operation report mode** attribute identifies the amount of output generated during operation execution.
- The **operation label state** attribute designates the operation to be performed on a labeled volume as; to be recycled or to be unlabeled.
- The **operation volume capacity** attribute specifies a volume's capacity.
- The **operation volume type** attribute specifies the types of volumes that may be considered when allocating a volume.
- The **operation ineligible** attribute specifies volumes ineligible for the current operation.
- The **operation task** attribute designates a secondary task or operation to be performed with the current operation.
- The **operation result** attribute reports error messages for multiple operations. This attribute maintains error messages for 32 simultaneous operations performed on a jukebox, that failed.
- The **operation instance** attribute specifies the instance number associated with an operation.
- The **operation next instance** attribute specifies the instance number associated with the next simultaneous operation.
- The **operation instances** attribute specifies the instance number for each simultaneous operation currently executing.
- The **operation hostname** attribute identifies the name of the system an operation is to executed.This attribute is used for jukeboxes who support devices, attached to multiple hosts, where the host machine may be inferred from other attributes, such as **operation device.**
- The **operation template** attribute specifies the template a label operation will use.
- The **operation volume pool** attribute specifies the default volume pool for label operations.
- The **operation source pool** attribute specifies the pool a volume will be selected for recycling.

- The **operation uses left** attribute indicates the number of times a cleaning cartridge can used.
- The **volumes** attribute specifies the names of resident volumes in corresponding order to the slot number.
- The **volume ids** attribute specifies the volume identifiers (volid) for resident volumes.
- The **volume cartridge ids** attribute tracks the identifier for each cartridge h a volume resides.
- The **loaded volumes** attribute contains the names of volumes currently loaded in jukebox devices.
- The **loaded bar codes** attribute identifies the bar codes of loaded volumes.
- The **loaded slots** attribute identifies the slot numbers of loaded volumes.
- The **event tag** attribute specifies the tag of the last notification event sent to the **nsrd** service
- The **event message** attribute is the text of the last notification event sent to the **nsrd** service.
- The **messages** attribute specifies the log messages from previous operations `nsrjb` has completed.
- The **minimum space** attribute specifies the low water mark of the remaining space on the volumes contained in the available slots.
- The **jukebox options** attribute specifies the options for this jukebox.
- The **auto clean** attribute specifies automatic cleaning for each device.
- The **cleaning slots** attribute identifies the range of slots in a jukebox that have been set aside for cleaning cartridges. For a pair of slot numbers the first number of the pair is less than or equal to the second. When **auto clean** is set to **yes** the range of slots specified for this attribute are assumed to contain cleaning cartridges, and the range of slots specified by **available slots**.
- The **default cleanings** attribute specifies the number of uses assigned to a new cleaning cartridge during an inventory of a jukebox by `nsrjb`.
- The **auto media management** attribute indicates whether automated media management for a jukebox is enabled. If the value is set to **yes**, unlabeled volumes in a jukebox may be automatically labeled.
- The **STL device names** attribute specifies silo device names of the devices identified in the **devices** attribute of a Silo Tape Library.
- The **STL interface lib** attribute indicates the path name of the dynamically linked Silo Tape interface library.
- The **STL device sharing** attribute specifies, how device sharing is handled. Device sharing is the automatic, load dependent, device switching for devices within a Silo Tape Library between different connected hosts. When this attribute is specified as **perm-max**, **perm** and **max** are numbers with **perm** < **max**, and **perm** is the number of devices, which can be reserved permanently.
- The **STL barcodes** attribute indicates the barcodes of the volumes residing within in a Silo Tape library, which are available to Backup.
- The **STL device reservation** attribute specifies the reservation state of shared devices in a Silo Tape library.
- The **allocated devices** attribute specifies jukeboxes allocated to RLM.

- The **application name** attribute specifies the name used by a server to identify itself to OpenVault when submitting a request to access resources on a jukebox.
- The **application key** attribute specifies the key used by a Backup server to identify itself to OpenVault when submitting a request to access resources on jukebox.
- The **jukebox lock** attribute synchronizes access to resources in a jukebox that supports multiple simultaneous operations. This attribute can be used to lock and unlock a entire jukebox.
- The **device locks** attribute synchronizes access to device resources in a jukebox that supports multiple simultaneous operations. The first two numbers of this attribute identify a range of devices locked, and the third number is the instance number assigned to the lock operation.
- The **volume/slot locks** attribute synchronizes access to volume and slot resources in a jukebox. The first two numbers of this attribute identifies the range of volumes/slots locked and the third number is the instance number assigned to the operation holding the lock.

Following is an example of a **NSR jukebox** resource named `Huntington`*:*

```
     type:NSR jukebox;
     name:Huntington;
     model:EXB-210;
     physical slots:1-11;
     control port:scsidev@0.6.0;
     devices:c:\dev\rmt\0mbn, c:\dev\rmt\1mbn;
     number device:2;
     write enabled:Yes;
     bar code reader:Yes;
     match bar code labels:Yes;
     volume expiration:;
     available slots:2-11;
     enabler code:012345-6789ab-cdef00;
     operation:Load;
     operation device:h:\dev\rmt\0mbn;
     operation slots: 1-10;
     operation options:manual;
```

```
    operation barcodes: A01B, A0/3-5/B;
    operation response:Yes;
    operation report mode:verbose;
    operation label state:recycle;
    operation volume capacity:10G;
    operation volume type:8mm, dlt;
    operation ineligible:;
    operation task: mount after label;
    operation instance:3;
    operation next instance:2;
    operation hostname:host1;
    operation template:default;
    operation volume pool:NonFull;
    operation source pool:Default;
    volumes:venus.001, venus.002, venus.003;
    volume ids:24198, 24199, 24200;
    STL device sharing:2-4;
    STL device reservation:;
    STL interface lib:h:\usr\lib\libstl.sol;
    event tag:6319962287;
    event message:could not unload device h:\dev\rmt\1mbn;
    messages:“09/12/97 11:50:56 CREATED”;
    minimum space:7g;
    jukebox options:two_sided;
    auto clean:Yes;
    cleaning slots:1;
    default cleanings:12;
    auto media management:Yes;
    reset class:initialize unload;
    jukebox lock:10;
    device locks:1-1-10;
    volume/slot locks:1-5-10;
```

## NSR Label

The **NSR label** resource describes the templates for generating volume labels. To edit the **NSR label** resources for a Backup server use `nsradmin` or use the Backup Administrator GUI (`nwadmin`).

The **NSR label** resource has the following attributes:

- The **name** attribute specifies the name of a label template.
- The **fields** attribute specifies constituent fields of a label template. When generating a volume name, the current value of each field is concatenated. If a *separator* is defined, they are placed between fields to form a volume name.

The types of fields are: numeric range, lower-case range, upper-case range and a list of strings. Each fields position is indicated by the *next* attribute.

- The **separator** attribute specifies a character separator for field labels.
- The **next** attribute specifies the next volume name to use. After a name is assigned to a volume, the next volume name will be generated and placed here.

The following is an example of a **nsr_label** resource:

```
type:NSR label;
    name:engineering;
    fields:aa-zz, 00-99;
    separator:.;
    next:aa.00;
```

## NSR License

The **NSR license** resource describes the features enabled in your Backup installation. To inspect the **NSR license** resources for a Backup server use `nsradmin` or use the Backup Administrator GUI (`nwadmin`).

The **NSR license** resource has the following attributes:

- The **name** attribute specifies the name of the license resource.
- The **enabler code** attribute specifies the code entered into the **nsrcap** command to enable the feature named in this resource.
- The **host id** attribute specifies the unique host id associated with the computer or licensed operating system.
- The **expiration date** attribute specifies the date an enabler will expire, if the enabler is an evaluation enabler or un-registered license enabler.
- The **auth code** attribute permanently authorizes an enabler. An unique, valid authorization code for an enabler is obtained from SunSoft through the registration of each purchased license enabler.

---

**Caution –** If a server’s host ID changes, all **auth codes** will immediately be invalidated, and the enablers must be re-registered with Sun to obtain new authorization codes.

---

- The **license type** attribute describes the specific feature(s) enabled.
- The **checksum** attribute maintains consistency of a NSR license resource, and between license resources.

Following is an example of a **NSR license** resource:

```
type: NSR license;
name: Backup Advanced/10;
enabler code: 123456-123456-123456;
host id: 7260d859;
expiration date: Authorized - no expiration date;
auth code: abcdef00;
license type: B10;
checksum: xxxxxxxxxxxxxxxxxxxxxx;
```

## NSR migration

The **NSR migration** resource specifies the files to be saved, the schedule, directives to use to omit files from a save, the group files will be pre-migrated with, the high-water and low-water marks to use for migration, the minimum access time and file size for migration, a list of file owners and groups to include or exclude during migration, and a list of file name patterns to skip.

To edit the **NSR migration** resources for a Backup server use `nsradmin` or use the Backup Administrator GUI`nwadmin`.

The **NSR migration** resource has the following attributes:

- The **name** attribute identifies the Backup client and save set whose migration attributes are stored in this resource.
- The **client** attribute identifies the HSM client whose save sets are to be placed under migration control.
- The **save set** attribute specifies the path names of filesystems or sub-trees to place under migration control for the specified client.
- The **enabled** attribute specifies whether a save set named in a resource will be automatically migrated.
- The **directive** attribute indicates to the client how to migrate certain files. The choices are defined by the existing directives.
- The **group** attribute indicates the groups a client or saveset is a part of for pre-staging migrated files.
- The **highwater mark %** attribute specifies the point at which files will start being replaced by stubs, measured as a percentage of available space used on a file system.
- The **low water mark %** attribute specifies the point at which files will stop being replaced by stubs, measured as a percentage of available space used on the file system.
- The **last access time** attribute specifies those files that have not been accessed in the past specified relative time will be migrated.
- The **minimum file size (KB)** attribute indicate files that are larger than then a specified size, will be migrated.

- The **file owner** attribute specifies the users whose files to be migrated.
- The **file group** attribute specifies a groups whose files are to be migrated.
- The **preserve** attribute indicates regular expressions, in a client’s shell syntax.
- The **statistics** attribute specifies statistics about recent migration activity for save set(s) managed using a resource.
- The **update statistics** attribute controls whether statistics in this resource should be updated to match the current values on a client.

Following is an example of a **NSR migration** resource defining an HSM client, called `elantra`:

```
type: NSR migration;
name: “elantra:c:\test”;
client: elantra;
save set: c:\test;
enabled:  Yes;
directive:  Unix with compression directives ;
group:  Default;
high water mark (%): 90;
low water mark (%): 80;
last access time: ;
minimum file size (KB): 5;
file owner: joe, dave;
file group: staff, developers;
preserve: *.exe *.dll;
```

## NSR notification

The **NSR notification** resource is used for each combination of an event, priority, and action handled by the Backup notification system. A Backup notification consists of a single event type, a single priority, and a message. The notification system posts each message to the action of each **NSR notification** resource that includes an event type and priority. To edit the **NSR notification** resources for a Backup server use `nsradmin` or use the Backup Administrator GUI`nwadmin`.

The **NSR notification** resource has the following attributes:

- The **name** attribute specifies the name of a notification resource.
- The **event** attribute specifies a class of events that will trigger a given notification. The valid classes are:
    - **Media**, identifies events related to a media multiplexor subsystem
    - **Savegroup**, identifies events generated by `savegroup`
    - **Index**, identifies events related to the on-line file index subsystem, **Registration**, identifies events caused by changes in a product’s registration status

    - **Server**, identifies Backup server events, such as restarting.
- The **priority** attribute specifies the priority at which a notification will be triggered. The valid values in increasing priority order are:
    - **Info**, supplies information about the current state of a server
    - **Notice**, an important piece of information
    - **Warning**, gives information about a non-fatal error
    - **Waiting**, indicates the server is waiting for a routine task
    - **Critical**, the server detected an error condition that requires attention
    - **Alert**, a severe error condition that demands immediate attention
    - **Emergency**, a severe condition that may cause Backup to fail.
- The **action** attribute indicates a command line to be executed when a given event occurs.

Following is an example of a **NSR notification** resource:

```
type: NSR notification;
name: savegroup completion;
administrator: root;
action: h:\usr\ucb\mail -s savegroup completion;
event: Savegroup;
priority: Info, Notice, Warning, Waiting;
```

## NSR policy

The **NSR policy** resource controls how long entries remain in a client's online file index, and when to mark a save set as recyclable. Each **NSR client** resource uses two policies, a browse policy and a retention policy. Each policy defines an amount of time determined by the *period* and the *number of periods*.

To edit the **NSR policy** resources for a Backup server use `nsradmin` or use the Backup Administrator GUI`nwadmin`.

The **NSR policy** resource has the following attributes:

- The **name** attribute specifies the name of the policy defined by this resource. This name will appear as an option of each NSR client resource.
- The **period** attribute indicates the base unit for a policy as one of the following values:
    - **Weeks**, defined as 7 days
    - **Months**, defined 31 days
    - **Years,** defined as 366 days.
      *Example:* period: Months;
- The **number of periods** attribute specifies the number of base units to use.

Following is an example of a **NSR policy** resource named `Quarter`:

```
type: NSR policy;
name: Quarter;
period: Months;
number of periods: 3;
```

## NSR Pool

The **NSR pool** resource describes each Backup pool, that determines a save sets browse and retention policies.This resource determines where volumes save sets reside based upon their characteristics.

There are four types of pools:

- **Backup** pools accept data from `savegroup` and manual backups.
- **Archive** pools accept archive data.
- **Backup clone pool**, where data from a backup pool can be cloned to.
- **Archive clone pool**, where archive data can be cloned to.

There are four pre-enabled pools shipped with Backup:

- **Default pool**, collects any backup data that is not directed to a customized pool.
- **Archive pool**, collects any archive data not directed to a customized pool.
- **Default clone pool**, is available to clone backup data to.
- **Archive clone pool**, is available for users to clone archive data to.

There are also a few pools shipped with Backup that are not enabled by default:

- Use the **Full** and **NonFull** pools, to segregate full level backups from other backups, for example, fulls versus incrementals.
- Use the **Offsite**, pool to generate offsite backups, as index entries are stored for the media pool and will not be referenced during normal recovers.

To edit the **NSR pool** resources for a Backup server use `nsradmin` or use the Backup Administrator GUI (`nwadmin`).

The **NSR pool** resource has the following attributes:

- The **name** attribute specifies the name of pool resources used when labeling volumes and determines which volumes a save set will reside.
- The **groups** attribute specifies the groups allowed in a pool.
- The **clients** attribute specifies the clients allowed in a pool. If a group is specified, clients that are members of that group can be listed.
- The **save sets** attribute indicates the save sets allowed in a pool. Save sets can be matched using regular expression matching.
- The **levels** attribute specifies the levels allowed in the specified pool.
- The **archive only** attribute enables archive only saves for a pool.
- The **status** attribute indicates the status of a pool as one of the following:

  - **enabled**, the pool is considered for determining what pools a save set should be saved to when performing backup volume selection.
  - **clone**, this pool is considered as the destination for cloning.
  - **disabled**, this pool is completely ignored.
- The **label template** attribute specifies the label template referenced when generating volume names for a pool.
- The **devices** attribute indicates a devices volumes within this pool that are allowed to be mounted to.
- The **store index entries** attributes specifies the entries made into a file index for backups. If entries are not made into the file index e, only media database entries for the save sets will be created.
- The **auto media verify** attribute will verify data written to volumes from this pool. Data is verified by re-positioning the volume to read a portion of the data previously written to the media and comparing the data read to the original data written. If the data read matches the data written, verification succeeds otherwise it fails.
- The **recycle to other pools** attribute specifies whether or not a given pool allows other pools to recycle its recyclable volume for their use.
- The **recycle from other pools** attribute specifies whether a given pool can recycle volumes from other pools when it exhausts all of its write-able and recyclable volumes.
- The **volume type preference** attribute specifies the selection factor made when their is a request for a write-able volume. The preferred type will be considered first within a priority level such as *jukebox* or *stand alone device.*

Following is an example of a **NSR pool** resource:

```
type:NSR pool;
archive only:No;
clients:;
devices:;
groups:;
label template:Default;
levels:;
name:Default;
save sets:;
status:Enabled;
store index entries:Yes;
auto media verify:Yes;
recycle from other pools:Yes;
recycle from other pools:Yes;
volume type preference:4mm;
```

## NSR Schedule

The **NSR schedule** resource describes a sequence of levels controlling the amount of data saved by Backup clients. There is one **NSR schedule** resource for each Backup schedule.

To edit the **NSR schedule** resources for a Backup server use `nsradmin` or use the Backup Administrator GUI (`nwadmin`).

The **NSR schedule** resource has the following attributes:

- The **name** attribute specifies a schedule’s name used by a client.
- The **period** attribute specifies the length of a schedule. It may be either “Week” or “Month.”
- The **action** attribute specifies the sequence of save levels within a schedule. One entry is used for each day of a schedule. The valid levels are **full**, **incr**, **skip**, and the numbers **1** through **9**. When the action attribute does not account for every day in the period, Backup will repeat the list of actions when the end of the action list is reached.
- The **override** attribute specifies a list of actions and dates overriding the actions specified in the **action** attribute. The format of an override specification is **action date**.

Following is an example of a **NSR schedule** resource:

```
type:NSR schedule;
     name:quarterly;
     period:Month;
     action:5 incr incr incr 9 incr incr;
     override:f 1/1/1997, f 3/1/1997;
```

## NSR Stage

The **NSR Stage** resource describes the staging policy used by a Backup server. To edit the **NSR Stage** resources for a Backup server use `nsradmin` or use the Backup Administrator GUI (`nwadmin`).

The **NSR stage** resource has the following attributes:

- The **name** attribute specifies the staging policy name.
- The **enabled** attribute specifies whether or not save sets are automatically staged from devices associated with a policy. It also enables and disables the periodic recover space operations.
- The **max storage period** attribute specifies the maximum number of days for a save set in a given volume before it is staged to a different volume.
- The **high water mark %** attribute specifies the point at which save sets should be staged, measured as the percentage of available space used on the file system. Staging will continue until the lower mark is reached.

- The **low water mark %** attribute specifies the point at which the staging process should stop, measured as the percentage of available space used on the file system.
- The **Save set selection** attribute specifies the save set selection criteria for staging. It may be one of four values:
  - largest save set
  - smallest save set
  - oldest save set
  - youngest save set
- The **Destination pool** attribute specifies the pool save sets should be sent.
- The **Devices** attribute specifies the *file type* devices are associated with.
- The **Recover space interval** attribute specifies the number of hours between recover space operations for save sets with no entries in the media database form file devices.
- The **Fs check interval** attribute specifies the number of hours between file system check operations.
- The **Start now** attribute specifies the selected operation to be triggered immediately on all devices associated with a policy. Operation can be one of the following:
  - **Check fs**, check file system and stage data if necessary.
  - **Recover space,** recover space for save sets with no entries in the media database.
  - **Stage all save sets**, stage all save sets to the destination pool.

Following is an example of a **NSR Stage** resource:

```
type: NSR stage;
name: test stage1;
autostart: Enabled;
max storage period: 7;
high water mark (%): 90;
low water mark (%): 85;
save set selection: largest save set;
destination pool: Default Clone;
devices:h:\disk\fd0;
start now: ;
```

## NSR

The **NSR** resource describes a Backup server and its clients. Each resource represents a component of a Backup system that needs administration. Resources are manipulated to control a Backup system. The file and the resources in them are accessible through the **nwadmin** and **nsradmin** programs, and can be viewed with a text editor.

Each resource is described by a list of attributes. Each attribute consists of a name and optional list of values. The attribute name is separated from an attributes options by a colon (**:**), attribute values are separated by commas (**,**), and each attribute ends in a semicolon (**;**). A comma, semicolon or back-slash (\) at the end of a line continues the line.

Following is an example of a **resource**, with eight attributes.

```
 type: NSR client;
         name: venus;
       server: earth;
     schedule: Default;
    directive: Unix standard directives;
        group: Default;
     save set: All;
remote access: ;
```

Each NSR resource includes the following attributes:

- The **type** attribute defines the attributes a resource can contain.
- The **name** attribute specifies the descriptive name of an object that a resource represents.
- The **administrator** attribute specifies the users that can modify or delete a resource. This attribute is inherited from the **type: NSR** resource when a new resource is created.
- The **hostname** attribute specifies the hostname of the system where a service that controls the specified resource is running.
- The remaining attributes (**ONC program number**, **ONC version number**, and **ONC transport**) specify the Open Network Computing information for a service.

Backup defines the following types of resources:

- The **NSR** resource describes a Backup server. It contains attributes that control administrator authorization, information about operations in progress, and statistics and error information about past operations.
- The **NSR client** resource describes a Backup client. It includes attributes that specify the files to save, which schedule to use, and which group this client belongs to.
- The **NSR device** resource describes a storage device. It includes attributes that specify a particular device name, media type, and name of the currently mounted volume.

- The **NSR directive** resource describes a directive. Directives control how a client's files are processed as they are being saved.
- The **NSR group** resource specifies a logical grouping of Backup clients and a backup starting time.
- The **NSR jukebox** resource describes a jukebox. It includes attributes such as the jukebox model, the first and last slot numbers in the jukebox, and the names of the devices within the jukebox.
- The **NSR label** resource specifies a template describing a sequence of names to be used when labeling volumes.
- The **NSR license** resource contains licensing information for each feature currently enabled. It contains various enabler and authorization codes used by Backup to validate licensed capabilities.
- **The NSR notification** resource specifies an action to be performed when a particular type of Backup event takes place.
- The **NSR policy** resource is used as part of the index management process. These policies control how long entries remain in a client's on-line file index and when to mark a save set as recyclable.
- The **NSR pool** resource is used by Backup to determine where volume save sets should reside on based on the characteristics of the save.
- The **NSR schedule** resource defines a sequence of save levels and an override list. The override list is made up of pairs of levels and dates. The level controls the amount of data saved when a client is backed up.

## Server Status Resource for Character-Based Displays

The `nsrwatch` program displays a Backup server's status from any system with enough `termcap` capabilities for cursor positioning. The `nsrwatch` program gets its information through remote procedure calls to the specified server. You can invoke `nsrwatch` from any machine that can access the Backup server through the network. If you do not specify a particular server, the server selection rules apply.

The `nsrwatch` display is divided into a header and several panels: the Server panel, the Device panel, the Sessions panel, the Messages panel, and the Pending messages panel. The panel sizes adjust depending on the size of the terminal or window used.

The header contains the name of the server and the current time. The Server panel provides information on the current status of the server (error messages, how long the server has been running, and the version of Backup software the server is using). The Device panel displays all the devices known to the Backup server. For each device, the panel displays the device type, the name of the currently mounted volume (or "unmounted" if there is none), and the device's status. If the device name has a J listed after it, the device resides in an autochanger or silo. The Sessions panel provides current save set information for each active session (save, recover, or browse). The Message panel displays a history of Backup messages of general interest to the operator. Finally, the Pending message panel displays messages that require operator intervention.

The nsrwatch program runs continuously until stopped by typing q or interrupted by a Control-Z or Control-C keystroke. If you type Control-L, the screen is cleared and refreshed with current information.

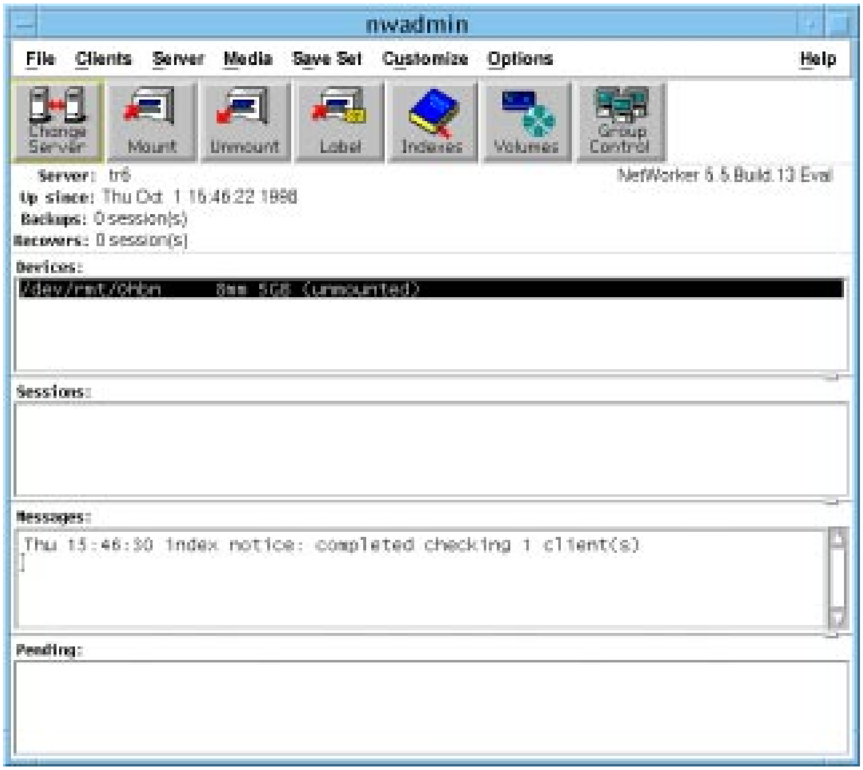
The following example describes the format and options available for the nsrwatch program:

```
nsrwatch [-s server] [-p polltime]
```

- Use the -s *server* option to specify a particular Backup server on the network.
- Use the -p *polltime* option to set the polling interval to be in *polltime* seconds.

# nwadmin

The nwadmin program is an X Window System application that is used to administer and monitor Backup servers. You can specify which Backup server to administer by using the -s option with the nwadmin command. If no server option is specified, nwadmin uses the server selection rules outlined under “User Interface Startup” on page 266. The main administration window is shown in FIGURE B-2 on page 297. Note that the value for Build displayed with the software version in the upper right corner of the window may differ from the example shown in the figure, depending on the final build of the software and whether you install patches for that version at a later date.



Server: tr6
NetWorker 5.5 Build.13 Eval
Up since: Thu Oct 1 15:46:22 1998
Backups: 0 session(s)
Recovers: 0 session(s)

```
/dev/rmt/0hbn      8mm 5GB (unmounted)
```

```
Thu 15:46:30 index notice: completed checking 1 client(s)
```

**FIGURE B-2** Administration Interface Provided by `nwadmin`

## Online Help for `nwadmin`

Online help is available through the Help menu in the main window and any subsequent windows, or through the Help button in any of the dialog boxes. There are four choices available from the Help menu:

- To view the online help available for any window displayed by the `nwadmin` program, select On Window from the Help menu.
- To view a list of topics for which online help is available, select On Topic from the Help menu and select a topic from the scrolling list that appears in the lower portion of the Help window.
- To view the online help topic that describes how to use the online help, select On Help from the Help menu.
- To view the version number of the software installed on the machine on which you invoked the `nwadmin` program on, select On Version from the Help menu.

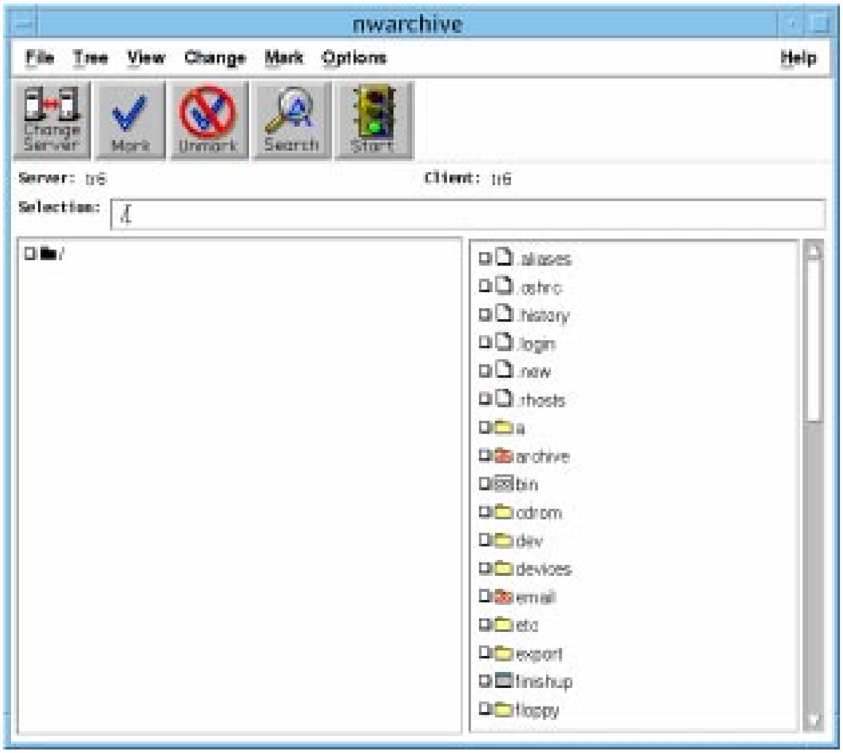
The following example describes the format and options available for the `nwadmin` program:

```
nwadmin [-s server]
```

- Use the `-s` *server* option to specify a particular Backup server on the network.

## nwarchive

The `nwarchive` program is an X Window System application that provides a GUI to the `nsrarchive` program, which is used to archive files on a manual basis to a Backup server. You can specify which Backup server to send archived data to by using the `-s` option with the `nwarchive` command. If no server option is specified, `nwarchive` uses the server selection rules outlined under "User Interface Startup" on page 266. The main window for `nwarchive` is shown in the following figure.



Server: tr6

Client: tr6

Selection: /

**FIGURE B-3** Main Window for `nwarchive` Program

## Online Help for `nwarchive`

Online help is available through the Help menu in the main window and any subsequent windows, or through the Help button in any of the dialog boxes. There are four choices available from the Help menu:

- To view the online help available for any window displayed by the `nwarchive` program, select On Window from the Help menu.
- To view a list of topics for which online help is available, select On Topic from the Help menu and select a topic from the scrolling list that appears in the lower portion of the Help window.
- To view the online help topic that describes how to use the online help, select On Help from the Help menu.
- To view the version number of the software installed on the machine that you invoked the `nwarchive` program on, select On Version from the Help menu.

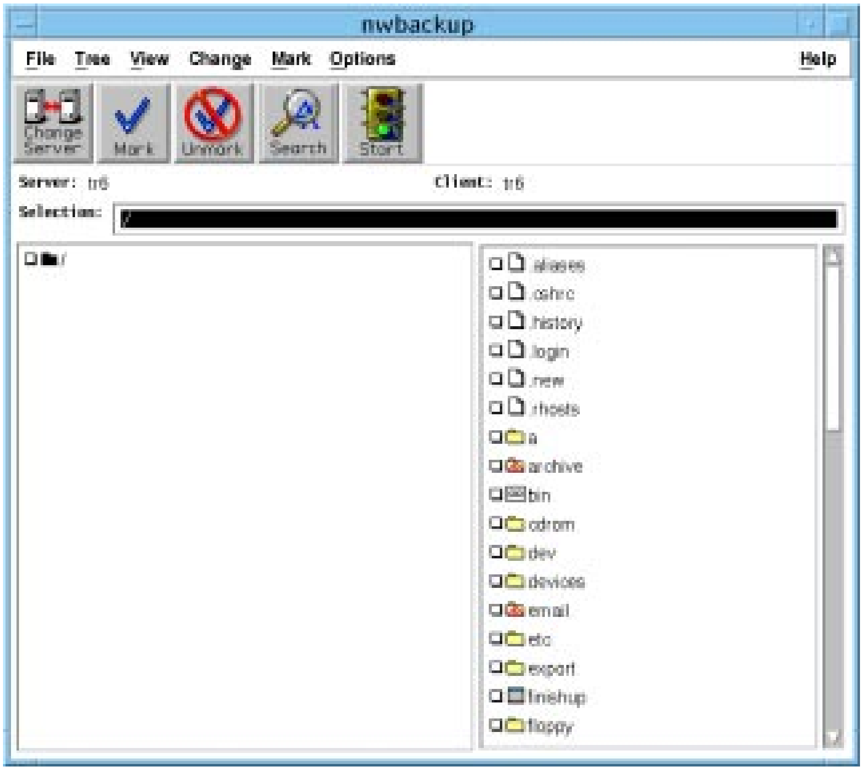
The following example describes the format and options available for the `nwarchive` program:

```
nwarchive [-s server]
```

- Use the `-s` *server* option to specify a particular Backup server on the network.

# nwbackup

The `nwbackup` program is an X Window System application that provides a GUI to the `save` program, and is used to perform a manual backup initiated from the client rather than from the server's `savegrp` program. You can specify which Backup server to use for the backup by using the `-s` option with the `nwbackup` command. If no server option is specified, `nwbackup` uses the server selection rules outlined under "User Interface Startup" on page 266. The main window for `nwarchive` is shown in the following figure.



Server: tr6

Client: tr6

Selection: /

**FIGURE B-4** Main Window for `nwbackup` Program

## Online Help for `nwbackup`

Online help is available through the Help menu in the main window and any subsequent windows, or through the Help button in any of the dialog boxes. There are four choices available from the Help menu:

- To view the online help available for any window displayed by the `nwbackup` program, select On Window from the Help menu.
- To view a list of topics for which online help is available, select On Topic from the Help menu and select a topic from the scrolling list that appears in the lower portion of the Help window.
- To view the online help topic that describes how to use the online help, select On Help form the Help menu.
- To view the version number of the software installed on the machine that you invoked the `nwbackup` program on, select On Version from the Help menu.

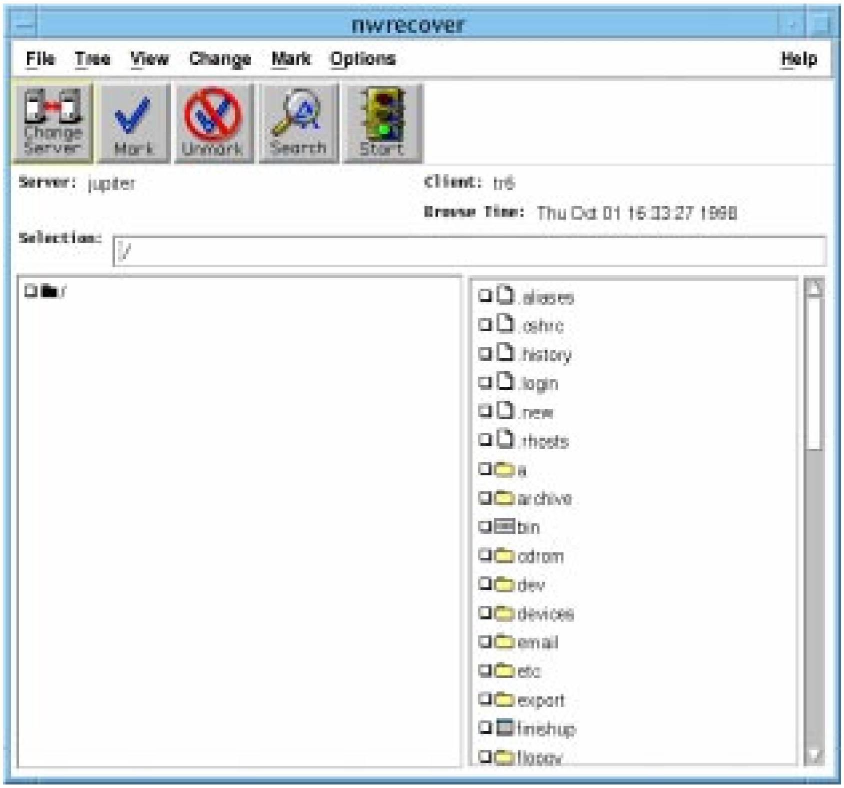
The following example describes the format and options available for the nwbackup program:

```
nwbackup [-s server]
```

- Use the -s *server* option to specify a particular Backup server on the network.

## nwrecover

The nwrecover program is an X Window System application that is used to administer and monitor Backup servers. You can specify which Backup client to recover data to by using the -c option with the nwrecover command. You can also specify which Backup server to recover the data from by using the -s option with the nwrecover command. If no server option is specified, nwrecover uses the server selection rules outlined under “User Interface Startup” on page 266. The main window for nwrecover is shown in the following figure.



Server: jupiter

Client: tr6

Browse Time: Thu Oct 01 16:33:27 1998

Selection: /

```
/
```

```
.aliases
.cshrc
.history
.login
.new
.rhosts
a
archive
bin
cdrom
dev
devices
email
etc
export
finishup
floppy
```

FIGURE B-5 Main Window for `nwrecover` Program

## Online Help for `nwrecover`

Online help is available through the Help menu in the main window and any subsequent windows, or through the Help button in any of the dialog boxes. There are four choices available from the Help menu:

- To view the online help available for any window displayed by the `nwrecover` program, select On Window from the Help menu.
- To view a list of topics for which online help is available, select On Topic from the Help menu and select a topic from the scrolling list that appears in the lower portion of the Help window.
- To view the online help topic that describes how to use the online help, select On Help from the Help menu.
- To view the version number of the software installed on the machine that you invoked the `nwrecover` program on, select On Version from the Help menu.

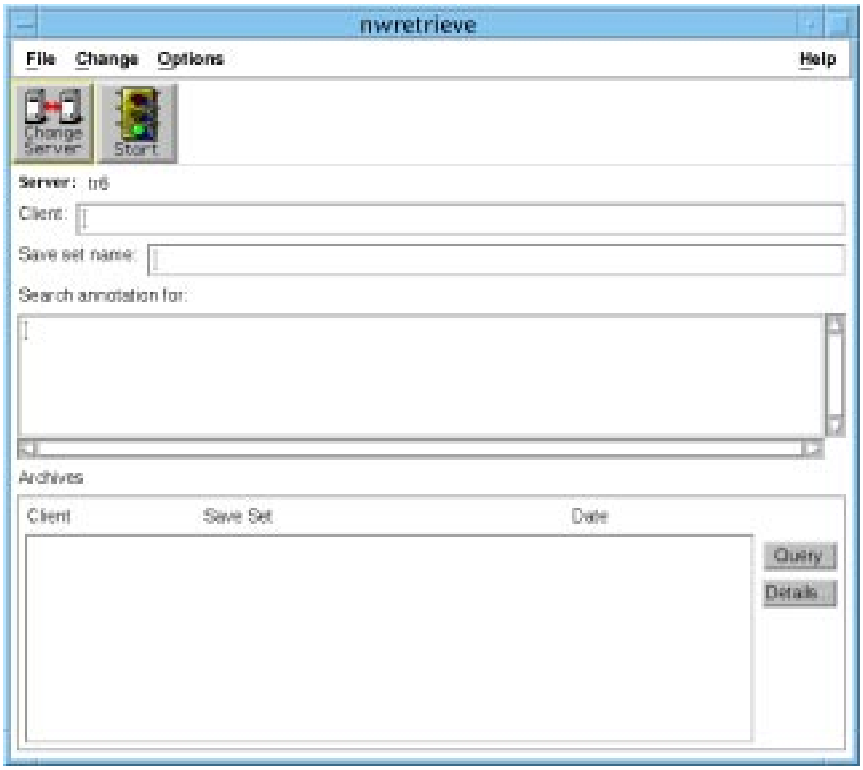
The following example describes the format and options available for the nwrecover program:

```
nwrecover [-c client] [-s server]
```

- Use the -s *server* option to specify a particular Backup server on the network.
- Use the -c *client* option to specify a particular Backup client on the network.

# nwretrieve

The nwretrieve program is an X Window System application that provides a GUI to the nsrretrieve program, which is used to retrieve archived files on a manual basis from a Backup server. You can specify which Backup server to retrieve the archived data from by using the -s option with the nwretrieve command. If no server option is specified, nwretrieve uses the server selection rules outlined under "User Interface Startup" on page 266. The main window for nwretrieve is shown in the following figure.



Server: tr6

**FIGURE B-6** Main Window for `nwretrieve` Program

## Online Help for `nwretrieve`

Online help is available through the Help menu in the main window and any subsequent windows, or through the Help button in any of the dialog boxes. There are four choices available from the Help menu:

- To view the online help available for any window displayed by the `nwretrieve` program, select On Window from the Help menu.
- To view a list of topics for which online help is available, select On Topic from the Help menu and select a topic from the scrolling list that appears in the lower portion of the Help window.
- To view the online help topic that describes how to use the online help, select On Help from the Help menu.
- To view the version number of the software installed on the machine that you invoked the `nwretrieve` program on, select On Version from the Help menu.

The following example describes the format and options available for the `nwretrieve` program:

```
nwretrieve [-s server]
```

- Use the `-s` *server* option to specify a particular Backup server on the network.

# Device and Media Management

This section provides a reference of the Backup command lines to use for device and media management. Some of the commands pertain specifically to the devices contained in an autochanger or silo; some commands apply specifically to SCSI devices, either standalone or in an autochanger.

The SCSI device library is a set of interfaces that Backup uses to communicate with SCSI devices. The SCSI devices are named in a platform-independent manner. The name assigned to the SCSI device is essentially a combination of *b.t.l*, where b is the logical SCSI bus, t is the SCSI target, and l is the SCSI *logical unit number* (lun) on that target.

A logical SCSI bus number may not be related to any specific platform hardware bus number; it may be a dense positive integer address space, that persists from system reboot to system reboot if the system hardware configuration remains the same. Target and lun information is contingent on the attached SCSI peripheral devices and their settings. Some platforms may allow dynamic addition and removal of SCSI devices, but may require a flush of the cached device information.

Typically, if a device does not have a system driver, users have permission to send SCSI device library commands. If a device has a system driver (for example, a tape drive), system privileges are required to send a command.

## changers

The `changers` program lists the SCSI autochangers that are attached to the system.

The following example describes the format and options available for the `changers` program:

```
changers [-dv] [-a b.t.l]
```

- Use the `-d` option to determine the names and addresses of the autochanger's media elements (for example, tape drives).
- Use the `-v` option to list more detailed information about each autochanger. The details provided may indicate how many media transports (MT), storage transports (ST), import/export elements (IE), and data transport (DT) elements the autochanger contains. The `-v` option also provides information about the element movement matrix supported by the autochanger.
- Use the `-a` option to identify a specific ordinal SCSI address for which you want to list information.

# hpflip

The `hpflip` program reads a Vendor Unique mode page from an HP Optical disk drive and toggles or "flips" the device type between OPTICAL and DIRECT ACCESS. Typically, most systems include drivers that can deal with removable DIRECT ACCESS device types (which are often limited to 512 byte/sector formatted disks). Systems with these device types often do not also have device drivers for OPTICAL device types. The `hpflip` program enables you to control how an HP Optical Disk Drive reports itself, and thus makes the OPTICAL device type available where it otherwise would have required an additional device driver.

The following example describes the format and options available for the `hpflip` program:

```
hpflip -a b.t.l [-r]
```

- You must use the required `-a` *b.t.l* argument to select a specific ordinal SCSI address, where *b* is the logical SCSI bus, *t* is the SCSI target, and *l* is the SCSI lun on that target.
- Use the `-r` option to reset the named device to OPTICAL, regardless of its current state. If you do not specify the `-r` option, the device type simply changes to the opposite of the current state.

# ielem

The `ielem` program sends an INITIALIZE ELEMENT STATUS command to the named SCSI device.

The following example describes the format and options available for the `ielem` program:

```
ielem -a b.t.l [-r element-address.number-of-elements]
```

- You must use the required `-a` *b.t.l* argument to select a specific ordinal SCSI address, where *b* is the logical SCSI bus, *t* is the SCSI target, and *l* is the SCSI lun on that target.
- If your autochanger supports the Vendor Unique EXABYTE autochanger `INITIALIZE ELEMENT STATUS` command, use the `-r` option to initialize the element status for a range of elements. Specify the starting element's decimal address and the number of elements whose status you want to read.

# inquire

The `inquire` program (in `/etc/LGTOuscsi` on Solaris systems) lists SCSI devices available. The `inquire` program returns INQUIRY data either for the named SCSI device (with the `-a` option) or for all SCSI devices attached to the system.

The following example describes the format and options available for the `inquire` program:

```
inquire [-c] [-a b.t.l]
```

- Enter the optional `-a` *b.t.l* argument to select a specific ordinal SCSI address, where *b* is the logical SCSI bus, *t* is the SCSI target, and *l* is the SCSI lun on that target.
- Enter the optional `-c` argument to force an INQUIRY command to be sent (to avoid picking up cached data that may be stale).

The `inquire` program always uses the built-in system drivers to test SCSI devices. The device type or pathname printed by the `inquire` program may be incorrect for devices that require special, third-party drivers.

**Caution –** The `inquire` program is not supported on HP-UX systems.

# jb_config

The `jb_config` program provides an interactive script for you to configure an autochanger resource for use with Backup. To run the `jb_config` program, the `nsrd` service must be running on the Backup server or storage node.

The script pauses periodically for you to enter a response to a prompt. If you want to accept the default choice displayed in braces, simply press [Return]. If you want to enter a different value, type the entry and press [Return].

After you configure the autochanger, use the `nsrcap` command or the Registration window to enter the enabler code for your Autochanger Software Module. You must have a separate enabler for each autochanger you want to use with Backup.

# jbexercise

The `jbexercise` program tests the functionality of an autochanger. Before you can run the program, you must empty all contents of the autochanger except for media loaded in the first and last slots. These pieces of media are moved around the autochanger as part of the various tests performed by `jbexercise`.

There are two major tests of functionality: drives and slots. Typically, both the drive and slot tests are run. You can test individual component types with the `-d` (for drives) and `-s` (for slots) options. In addition, you can test specific components with the `-D` and `-S` options. When you use the `-D` and `-S` options, the only test that runs is on the specified component (for example, if you name a specific slot, the drives test is not run). For drives, replace *drive* with the logical address of the component. For slots, replace *slot* with the physical address.

Upon startup, the program queries for the nonrewinding pathnames of the drives found in the configuration of the autochanger. This query is not performed if you are using an autochanger that does not require media to be ejected from a device (for example, because the device has automatic ejection capabilities).

The first test moves the media from the first slot to each of the drives. No operator intervention is required.

The second test loads the media from various slots to the first drive. The default is to test the media in the first and last slots in the autochanger. If you test a specific slot, you must first load that slot with media.

The following example describes the format and options available for the `jbexercise` program:

```
jbexercise -m model -c control-port [-V vendor-type]
[CdsIv] [-D drive] [-S slot]
```

- Use the `-c` option to specify the control port for the `jbexercise` command to interface with the autochanger.
- Use the `-C` option to return the configuration of the autochanger without further testing.
- Use the `-d` option to test only the drives.
- Use the `-D` option to test only a specific drive.
- Use the `-I` option to inventory the autochanger without testing.
- Use the `-m` to specify an autochanger model. (To list the supported autochanger models, run the `jbexercise` command without any arguments to print the usage string.)
- Use the `-s` option to test only the slots.
- Use the `-S` option to test only a specific slot.
- Use the `-v` option to use the `jbexercise` command in verbose mode, which displays more detailed information.
- Use the `-V` option to specify a particular vendor ID.

## ldunld

The `ldunld` program sends a `LOAD` or `UNLOAD` command to the named tape device to load or unload media.

The following example describes the format and options available for the `ldunld` program:

```
ldunld {-u | -l} -a b.t.l
```

There are three command options:

- You must use the required `-a` argument to select a specific ordinal SCSI address.
- Use the `-l` option to load media into a device.
- Use the `-u` option to unload media from a device.

## libsji

The `libsji` program describes the Standard Jukebox Interface (SJI) Library. The location of the SJI library varies from platform to platform.

The SJI library is a public set of interfaces that Backup uses to communicate with jukeboxes. Generally, this library converts SJI commands (as formed by Backup) to the appropriate SCSI commands, but the underlying attachment to the jukebox is irrelevant to the function of this interface.

There are three entry points into the SJI library:

- **void * sji_open (char *** *device-name***)**

The sji_open entry point opens a channel to the SJI-compliant jukebox specified by *device-name*. A channel token of type void * is returned if successful, otherwise a NULL token is returned. You can express the device name as an ordinal SCSI type (for example, scsidev@b.t.l). The device name can also be a platform-specific style device name (for example, /dev/sjid1u1) for those platforms that do not use Sun device drivers.

- **int sji_cmd (void *token, int cmd, void *arg)**

The sji_cmd entry point sends an SJI command to the device opened by sji_open.

- **void sji_close (void *token)**

The sji_close entry point closes a channel to the device opened by the call to sji_open.

The list of all the available commands and their arguments is too large to list here. Send e-mail to sji@Sun.com to request more information on these interfaces.

## lrescan

The lrescan program tells the underlying SCSI library to discard any cached information that it can and scan again for new devices.

## lreset

The lreset program tells the underlying SCSI library to reset the named logical SCSI bus. You must have administrative privileges to execute this command, which has the following format:

```
# lreset busnumber
```

**Caution –** The lreset command can cause the destruction of vital data, because the command causes a SCSI bus reset. The command may also crash your system. You should only use the lreset command as an extreme last resort to quit a process that is not responding.

# lusbinfo

The lusbinfo program prints out a limited amount of information about the SCSI buses attached to the system. If you use the optional -v argument, a verbose list of information about the devices in the attached SCSI buses is also printed. The following example shows the format to use for the lusbinfo program:

```
lusbinfo [-v]
```

# lusdebug

The lusdebug program sets a debug level for the underlying Backup SCSI device drivers. A debug level of 0 (zero) turns off debugging. Larger integers enable greater levels of debug information. If you enter an invalid debug level, the lusdebug program defaults to a debug level of zero. The following example shows the format to use for the lusdebug program:

```
lusdebug debug-level
```

# lusmode

The lusmode program prints a large amount of MODE information about the SCSI devices attached to the system.

# msense

The msense program sends a MODE SENSE command to the named SCSI device and is only indented as input to the pmode command.

The following example describes the format and options available for the msense program:

```
msense -a b.t.l. [-p pagecode]
```

- You must use the required -a *b.t.l* argument to select a specific ordinal SCSI address, where *b* is the logical SCSI bus, *t* is the SCSI target, and *l* is the SCSI lun on that target.

- Use the `-p` option to select a specific mode page. If you do not specify a specific mode page, all pages are fetched (code 0x3f). You must specify the *pagecode* argument in hexadecimal notation.

# nsrjb

The `nsrjb` program manages autochangers for Backup servers. Use the `nsrjb` command, rather than the `nsrmm` command, to label, load, and unload the volumes contained in an autochanger. Only one `nsrjb` command can access an autochanger at a time.

The `nsrjb` program attempts to determine which autochanger to use based on the options `-j`, `-f`, or a *volume name*. If one or more of these options do not uniquely identify an autochanger and one must be selected, the `nsrjb` program prompts you to choose an autochanger. Alternatively, you can set the NSR_JUKEBOX environment variable to the name of the autochanger you want the `nsrjb` program to use by default.

The following example describes the format and options available for the `nsrjb` program:

```
nsrjb [-C] [-j autochanger-name] [-v]
[-f media-device] [-S slots] [volume-name]
nsrjb -L [-j autochanger-name] [-gnqvM] [-R | -B]
[-Y | -N] [-b pool] [-f media device] [-e expire]
[-c capacity] [-o mode] [-S slots | -T tags]
[volume-name]
nsrjb -l [-j autochanger-name] [-nvqrMR]
[-f media-device] {-S slots | -T tags | volume-name}
nsrjb -u [-j autochanger-name] [-qvM]
[-f media-device] [-S slots -T tags] [volume-name]
nsrjb -I [-j autochanger-name] [-Ev]
[-f media-device] [-S slots | -T tags]
nsrjb -p [-j autochanger-name] [-v]
[-f media-device] [-S slots -T tags]
nsrjb -o mode [-j autochanger-name] [-Y]
{-S slots | media device}
nsrjb -H [-j autochanger-name] [-E] [-v]
nsrjb -h [-j autochanger-name] [-v]
nsrjb -U uses [-j] [-S slots | -T tags]
nsrjb -V [-j autochanger-name] [-v]
nsrjb -d [-j autochanger-name] [-v] [-S slots]
[-P port] [volume-name]
```

```
nsrjb -w [-j autochanger-name] [-v] [-S slots]
[-P port] [volume-name]
nsrjb -a [-j autochanger-name] [-v] -T tags
nsrjb -x [-j autochanger-name] [-v] -T tags
nsrjb -F [-j autochanger-name] [-v] -f media-device
```

- Use the `-b` option to specify the pool to which you want to assign the volume. If you omit this option, the volume is automatically assigned to the Default pool.
- Use the `-B` option to verify that the volume does not already have a readable Backup label. If you specify this option and the volume has a Backup label, the label operation is canceled and an error message is displayed.
- Use the `-c` option to override the volume’s default capacity.
- Use the `-C` option to display the current volumes in the autochanger and the associated devices. The `-C` option does not perform an actual inventory.
- Use the `-d` option to deposit (load into the jukebox) a cartridge from the cartridge access port (CAP).
- Use the `-e` option to override the default volume expiration date.
- Use the `-E` option to initialize element status for autochangers that provide this feature. You can use this option in conjunction with the `-I` or `-H` options.
- Use the `-f` option to specify a media device rather than the jukebox control port. Use the pathname of the media device displayed in the `NSR jukebox` resource. When more than one media device is configured for a jukebox, `nsrjb` selects the first available media device, by default. The default device can be overridden by using the `-f` option.
- Use the `-h` option to display the actions and results of the past 100 autochanger commands issued.
- Use the `-H` option to reset the autochanger hardware (and the Backup database that represents the autochanger) to a consistent state. The autochanger clears the transport, and then unmounts and unloads volumes from the drives to slots. An inventory is not done (see the `-I` option). If the autochanger senses that the inventory is out-of-date, it prints an appropriate message.
- Use the `-I` option to perform an inventory on the autochanger’s contents. The volumes in the specified slots are loaded into a device and their labels are read. Use this option to ensure that the mapping between slot number and volume name is correct. This option may take a long time to complete.

For jukeboxes that have the element status capability (for example, the EXB-120, EXB-60, or HP optical models), you can use the `-E` option in conjunction with the `-I` option to reinitialize the autochanger’s inventory state. The `-E` option increases the time it takes to inventory the autochanger, because the hardware must check every component, including all slots and drives, for the presence of media. You only need to use this option if you manually swap media in or out of an autochanger.

Volumes from slots that are reserved for cleaning cartridges are not loaded during the inventory. If your autochanger does not support the element status or barcode reader features, you must use the `-U` option to enter a cleaning cartridge into the autochanger’s inventory. If your autochanger does support either of these features, the cleaning cartridge is indicated in the inventory with the volume name “cleaning tape.”

- Use the `-j` option to specify a particular autochanger for the `nsrjb` program to use. The given name is the one that you assigned when you created the NSR jukebox resource for the autochanger. If you supply the `-j` option, the NSR_JUKEBOX environment variable is overridden.
- Use the `-l` option to load and mount a volume. You must also specify a volume name or slot number.
- Use the `-L` option to label the volumes in the specified slots. If you do not specify any slots, the range of slots described in the `NSR jukebox` resource for the autochanger is used. If the autochanger has a barcode label reader and you set the `NSR jukebox` resource attributes “barcode reader” and “match barcode labels,” the volume label is derived from the barcode label on the media, and the media barcode label will be stored in the Backup media database. If you set the `NSR jukebox` resource attribute “match barcode labels,” the volume label is derived from the label template, although the media barcode label is stored in the Backup media database so that it can be used during inventory operations. You cannot label volumes that are in slots reserved for cleaning cartridges.
- Use the `-M` option to send messages to the N daemons that report progress and errors. This option is used by `nsrd` when mounting, unmounting, and labeling volumes on behalf of `nsrmmd` requests, and is not normally used for manual requests. If the `-M` option is used with a manual run of `nsrjb`, then the command line should also specify the jukebox (with the `-j` or `-k` flag). This allows `nsrjb` to choose a jukebox from those available.
- Use the `-n` option, in combination with the `-l` option, to load a volume without mounting it. This allows the `nsrjb` program to control an autochanger that contains non-Backup volumes.
- Use the `-N` option, in combination with the `-LR` options, to tell `nsrjb` to skip the confirmation prompt. When Backup recycles volumes, you normally receive a prompt to confirm that it is okay to overwrite any volumes that Backup considers nonrecyclable.
- Use the `-o` option to set the mode of a volume or range of slots. Choose one of the following mode values: [not]recyclable, [not]read-only, [not]full or [not]manual. The [not]manual modes are the only valid modes when used with the `-l` option. If you do not give the `-Y` option, you are prompted to confirm the operation for each volume. See “`nsrim`” on page 358 for a discussion of the per-volume flags.
- Use the `-p` option to verify and print a volume’s label.
- Use the `-P` option to specify the CAP to load or unload a volume from.
- Use the `-q` option to run the `nsrjb` program in quiet mode. You can only use this option in conjunction with the `-L`, `-l`, and `-u` options.
- Use the `-R` option to recycle the volumes. If a volume is recyclable, you are not prompted to confirm the recycle operation.
- Use the `-r` option to load a volume as read-only. You can only use this option in conjunction with the `-l` option.

- Use the `-S` option to specify a slot or range of slots to operate on. The `-l` and `-u` options only accept one slot: the other options accept a range of slots. Specify the slot range in low to high integer order. The range is checked for validity against the Jukeboxes resource that describes the autochanger. You can only specify one slot range at a time.
- Use the `-u` option to unload a volume from a device or slot.
- Use the `-U` option with the *uses* argument to set the number of times a cleaning cartridge may be used. You can use the `-T` option in conjunction with the `-U` option to add cleaning cartridges to a silo, which also reserves a slot in the silo for each cleaning cartridge added.
- Use the `-v` option to tell `nsrjb` to display verbose information about the commands executed.
- Use the `-V` option to display vendor-specific status information. When you combine the `-V` option with the `-v` option, the configuration of the autochanger is also displayed.
- Use the `-w` option to withdraw (unload from the jukebox) a cartridge to the CAP.
- Use the `-Y` option to disable the prompt for confirmation. If you issue the `nsrjb` command to relabel a volume that has an existing label and you do not use the -Y option, the nsrjb command fails, as the following example shows:

```
nsrjb -L -S 4
Are you sure you want to over-write 00000069 with a new label? y
nsrjb: Jukebox error, Fri 11:25:58 Will not over-write volume
without confirmation
```

The following `nsrjb` options are only valid for use with Silo Tape Libraries (STL):

- Use the `-a` option, in conjunction with the `-T` option, to allocate volumes in an STL for use by a Backup server or storage node. You must allocate a volume before you label it for Backup to use. You can add the `-d` option for silos that support the deposit (also known as importing or entering) of tapes through the silo's I/O port. The `-d` option must appear *after* the `-a` option on the command line. This function is usually handled by the silo management software, but is provided here for ease of use. The deposit option may not be supported on all the silos that Backup supports. See the `-x` option for a description of how the volumes are removed from an STL's list of volumes available for use by a Backup server.
- Use the `-F` option to release a shared device contained in an STL. This option is only available for tape libraries that support shared devices.
- Use the `-T` option to specify the tags or barcodes of volumes contained in an STL. You can specify a single volume tag or a volume tag template, which is similar to a regular Backup label template. The volume tag template consists of a list of template fields separated by slashes (/), whereas a Backup label template consists of an alphanumeric string or alphabetic or numeric range.
- Use the `-x` option, in conjunction with the `-T` option, to remove volumes from the STL's list of volumes available for use by a Backup server or storage node. You can add the `-w` option for silos that support the withdrawal or ejection of tapes

through the silo’s I/O port. The `-w` option must appear *after* the `-x` option on the command line. The silo management software usually handles this function, but it is provided here for ease of use. The withdrawal option may not be supported on all the silos that Backup supports. See the `-a` option for a description of how the volumes are allocated to an STL’s list of volumes available for use by a Backup server.

# nsrmm

The `nsrmm` program provides a command line interface to manage the media and backup devices used by Backup servers and storage nodes.

The following examples describe the format and options available for the `nsrmm` program:

```
nsrmm [-C] [-v | -q] [-s server] [-f device]
nsrmm -m [-v | -q] [-s server] [-f device] [-r]
[volume-name]
nsrmm -l [-v | -q] [-s server] [-f device] [-myB] [-e expiration] [-c
capacity] [-o mode] [-b pool] [-R | volume-name]
nsrmm {-u | -j} [-v | -q] [-s server] [-y]
[-f device | volume-name]
nsrmm -p [-v | -q] [-s server] [-f device]
nsrmm {-d | -o mode} [-v | -q] [-s server] [-Py] [S ssid[/cloneid] |
-V volume-id | volume-name...]
```

- Use the `-B` option to verify that the volume you want to label does not have a readable Backup label. If you specify this option and the volume has a valid Backup label, the label operation is canceled and an error message is displayed.
- Use the `-b` *pool* option to specify the pool to which the volume should be assigned. If you omit this option, the volume is automatically assigned to the Default pool. If you specify a pool name without specifying a volume name, the next volume name associated with the pool’s label template resource is used.
- Use the `-C` option to display a list of Backup-configured devices and the volumes currently mounted in them. The information is gathered from what the server inventory shows, and does not perform an actual volume operation, unlike the `-p` option described later. The `-C` option is the default.
- Use the `-c` option to override a volume’s default capacity. Backup normally uses built-in default capacities, based on the device’s type. The format of the specification is *number multiplier.* Number may be any value, including an integer or real number, with up to three decimal places. Multiplier may be one of “K” (1024 bytes), “M” (1000K), or “G” (1000M). Lowercase letters are acceptable, as are extra characters.

- Use the `-d` option to delete the client file indexes and media database entries from the Backup databases. This action does not destroy the data contained on the volume: instead, it removes all references used by Backup to the volume and the user files contained on it. You can use this option to control the size of the Backup databases.
- Use the `-e` *expiration* option to set the expiration date for volume relabel. This option overrides the default label expiration, which is two years. The value of expiration is entered in the format described in a special value of "forever" that is used for migration and archive volumes means that the volume label never expires.
- Use the `-f` device option to explicitly specify a device. When you configure more than one device, the `nsrmm` program selects the first device by default.
- Use the `-j` option to eject a volume from the device. This is similar to performing an unmount operation, except that the volume is also physically ejected from the device, if possible. This option is not available with many devices and media types.
- Use the `-l` option to label a volume for Backup to recognize and use. You must physically load the volume into the device, either by an operator or autochanger, before the label operation can proceed.
- Use the `-m` option to mount a volume in a device. The mount operation is performed after the volume is placed in the device and labeled; therefore, only labeled volumes can be mounted. You can combine the label and mount operation in one command line.
- Use the `-o` mode option to set the mode of a volume, save set, or save set instance (clone). Choose one valid mode value: [not]recyclable, [not]readonly, [not]full, [not]manual, or [not]suspect. The [not]recyclable mode applies to volumes or save sets, but not to clones. The [not]readonly, [not]full, and [not]manual modes are the only valid modes you can use with the `-l` option. The [not]suspect mode applies only to clones. You must specify the [not]suspect mode if you use the `-S` option with an SSID/cloneID specification. You do not need to specify the [not]suspect mode if you only specify ssid with the `-S` option. The suspect flag is set automatically when a `recover` operation encounters a media error when attempting to recover data from a particular save set clone.
- Use the `-P` option in conjunction with the `-d` option to purge the corresponding client file index entries, without deleting the entries in the media database. You can then use the `scanner` command to recover the file index entries.
- Use the `-p` option to verify and print a volume's label. When you use this option, mounted volumes are unmounted to verify the label.
- Use the `-R` option to relabel a volume. This option rewrites the volume's label and purges the client file index entries for all of the user files saved on the volume. Some of the volume usage information is maintained.
- Use the `-r` option to mount a volume as read-only. Volumes that are marked as full and volumes whose mode is set as read-only with the `-o` option are automatically mounted as read-only.
- Use the `-s` *server* option to specify the Backup server on which you want to invoke `nsrmm`.

- Use the -S *ssid* option with the -o option to change or the -d option to remove a save set from the Backup databases. The save set is specified by an SSID. A save set instance (clone) can only be specified with the -o option, using the format ssid/cloneid. You can use the mminfo program to determine the ssid and cloneid values.
- Use the -u option to unmount a volume. You should always unmount a volume before you unload it from a device.
- Use the -V *volid* option in conjunction with the -d option to remove a volume from the Backup server's media database. You can determine the value of the volume identifier (volid) with the mminfo program.
- Use the -v option to run the nsrmm program in verbose mode.
- Use the -y option to turn off confirmation of potentially destructive operations before nsrmm performs them. Use this option with extreme caution.

## pmode

The pmode program parses the data output by the msense program and prints the output in a format that you can read.

The following example describes the format and options available for the pmode program:

```
pmode [-f filename]
```

- Use the -f *filename* option to specify the input file to use for the pmode program (the file output from the msense program). If you do not specify the input, standard input is assumed.

The output from the pmode program is similar to the following:

```
mars# msense -a 0.0.0 -p 0x03 | pmode Mode Header: mdl=35 mtype=0x0
dparm=0x10 bdlen=8 Block Desc[0]: dens=0x0 nblks=3933040
blklen=512 Fixed Page, code 0x03
(Format Device): tracks_per_zone: 0xf
alt_sectors_per_zone: 0x22
alt_tracks_per_zone: 0x0
alt_tracks_per_vol: 0x0
sectors_per_track: 0x5e
data_bytes_per_sect: 0x200
interleave: 0x1
track_skew_factor: 0x8
cylinder_skew_factor: 0x11
SSEC: 0x0
HSEC: 0x1
RMB: 0x0
SURF: 0x0
```

# relem

The relem program sends a READ ELEMENT STATUS command to all changers, or to the (optionally, with the -a option) named device.

The following example describes the format and options available for the relem program:

```
relem [-a b.t.l] [-fvtb] [-m {0|1|2}]
[-r element-address.number-of-elements]
```

- Use the -a *b.t.l* option to select a specific ordinal SCSI address, where “b” is the logical SCSI bus, “t” is the SCSI target, and “l” is the SCSI lun on that target (for example, scsidev@0.4.0).
- Use the  -b option to have the returned element status data dumped as ASCII hexadecimal codes, rather than decoded information.
- Use the -f option to receive full, somewhat verbose output.
- Use the -m {0|1|2} option to indicate the method for obtaining element status data. If you specify -m 1, element status data is fetched for each element type (for example, all drive elements are read at once, then all slot elements, and so forth). If you specify the default method -m 2, element data is fetched on a per element basis.

- Use the `-r` *element-address.number-of-elements* option to read a range of addresses, where *element-address* is the starting decimal address (in the autochanger's numbering sequence) of the element to start from and *number-of-elements* is the number of elements of status to read.
- Use the `-t` option to print any volume tags encountered.
- Use the `-v` option to receive verbose output.

# sjidopen

The `sjidopen` program tests the SJIDOOROPEN command on SJI-compliant autochangers. The SJIDOOROPEN command tests the open/close capability of the main door to the autochanger. If an autochanger does not support this feature, an error message is returned. The following example shows the correct usage for the `sjidopen` program:

```
sjidopen device-name
```

The *device-name* option used with the `sjidopen` program represents any device name that can be used to reach an SJI-compliant autochanger driven by the system, typically in the form *b.t.l*, where *b* is the logical SCSI bus, *t* is the SCSI target, and *l* is the SCSI lun on that target (for example, `scsidev@0.4.0`).

# sjiielm

The `sjiielm` program tests the SJIIELEM command on SJI-compliant Jukeboxes. The SJIIELEM command tests the Initialize Element Status interface for an autochanger. If the autochanger does not support the element status feature, an error messages is returned. The following example shows the correct usage for the `sjiielm` program:

```
sjiielm device-name [{drive | slot | inlt | mt} address number-of-elements]
```

The *device-name* option used with the `sjiielm` program represents any device name that can be used to reach an SJI-compliant autochanger driven by the system, typically in the form *b.t.l*, where *b* is the logical SCSI bus, *t* is the SCSI target, and *l* is the SCSI lun on that target (for example, `scsidev@0.4.0`).

The additional options described next are for use with autochangers that support the initilization of a specific range of elements. If the autochanger supports this feature, select one of the following element types:

- drive

- slot
- inlt (import/export element)
- mt (media transport)

Specify the SJI normalized address (for example, starting from 1) and the number of elements to initilize.

## sjiinq

The `sjiinq` program tests the SJIINQ command on SJI-compliant autochangers. The SJIINQ command returns a string that identifies an autochanger. If the autochanger does not support this feature, an error message is returned. The following example shows the correct usage for the `sjiinq` program:

```
sjiinq device-name
```

The *device-name* option used with the `sjiinq` program represents any device name that can be used to reach an SJI-compliant autochanger driven by the system, typically in the form *b.t.l*, where *b* is the logical SCSI bus, *t* is the SCSI target, and *l* is the SCSI lun on that target (for example, `scsidev@0.4.0`).

## sjirdp

The `sjirdp` program tests the SJIRDP command on SJI-compliant autochangers. The SJIRDP command reads SJI ordinal device positions from an autochanger. The following example shows the correct usage for the `sjirdp` program:

```
sjirdp device-name
```

The *device-name* option used with the `sjirdp` program represents any device name that can be used to reach an SJI-compliant autochanger driven by the system, typically in the form *b.t.l*, where *b* is the logical SCSI bus, *t* is the SCSI target, and *l* is the SCSI lun on that target (for example, `scsidev@0.4.0`).

The following example represents typical output from the SJIRDP command:

```
scsidev@0.4.0 has 2 DATA TRANSPORT Elements starting at address 1
scsidev@0.4.0 has 1 MEDIA TRANSPORT Element starting at address 1
scsidev@0.4.0 has 25 STORAGE Elements starting at address 1
scsidev@0.4.0 has 1 IMPORT/EXPORT Element starting at address 1
```

# `sjirdtag`

The `sjirdtag` program tests the SJIRTAG command on SJI-compliant autochangers. The SJIRTAG command reads media presence and tag data from an autochanger. The following example shows the correct usage for the `sjirdtag` program:

```
sjirdtag device-name
```

The *device-name* option used with the `sjirdtag` program represents any device name that can be used to reach an SJI-compliant autochanger driven by the system, typically in the form *b.t.l*, where *b* is the logical SCSI bus, *t* is the SCSI target, and *l* is the SCSI lun on that target (for example, scsidev@0.4.0).

The following example represents typical output from the SJIRTAG command:

```
Tag Data for 0.4.0, Element Type DATA TRANSPORT:
Elem[001]: tag_val=0 pres_val=1 med_pres=1 med_side=0
Tag Data for 0.4.0, Element Type STORAGE:
Elem[001]: tag_val=0 pres_val=1 med_pres=1 med_side=0
Elem[002]: tag_val=0 pres_val=1 med_pres=1 med_side=0
Elem[003]: tag_val=0 pres_val=1 med_pres=1 med_side=0
Elem[004]: tag_val=0 pres_val=1 med_pres=1 med_side=0
Elem[005]: tag_val=0 pres_val=1 med_pres=0 med_side=0
Elem[006]: tag_val=0 pres_val=1 med_pres=1 med_side=0
Elem[007]: tag_val=1 pres_val=1 med_pres=1 med_side=0
VolumeTag=<00000098>
Tag Data for 0.4.0, Element Type MEDIA TRANSPORT:
Elem[001]: tag_val=0 pres_val=1 med_pres=0 med_side=0
```

# `sjirelem`

The `sjirelem` program tests the SJIRELEM command on SJI-compliant autochangers. The SJIRELEM command reads media presence and origin data from an autochanger. The following example shows the correct usage for the `sjirelem` program:

```
sjirelem device-name
```

The *device-name* option used with the `sjirelem` program represents any device name that can be used to reach an SJI-compliant autochanger driven by the system, typically in the form *b.t.l*, where *b* is the logical SCSI bus, *t* is the SCSI target, and *l* is the SCSI lun on that target (for example, `scsidev@0.4.0`).

The following example represents typical output from the SJIRELEM command:

```
Element Data for 0.4.0, Element Type DATA TRANSPORT:
Elem[001]: pres_val=1 med_pres=1 med_side=0
Origin: type STORAGE, address 5
Element Data for 0.4.0, Element Type STORAGE:
Elem[001]: pres_val=1 med_pres=1 med_side=0
Elem[002]: pres_val=1 med_pres=1 med_side=0
Elem[003]: pres_val=1 med_pres=1 med_side=0
Elem[004]: pres_val=1 med_pres=1 med_side=0
Elem[005]: pres_val=1 med_pres=0 med_side=0
Elem[006]: pres_val=1 med_pres=1 med_side=0
Elem[007]: pres_val=1 med_pres=1 med_side=0
Element Data for 0.4.0, Element Type MEDIA TRANSPORT:
Elem[001]: pres_val=1 med_pres=0 med_side=0
```

# sjirjc

The `sjirjc` program tests the SJIRJC command on SJI-compliant autochangers. The SJIRJC command reads internal configuration information and options about an autochanger and prints it out. The following example shows the correct usage for the `sjirjc` program:

```
sjirjc device-name
```

The *device-name* option used with the `sjirjc` program represents any device name that can be used to reach an SJI-compliant autochanger driven by the system, typically in the form *b.t.l*, where *b* is the logical SCSI bus, *t* is the SCSI target, and *l* is the SCSI lun on that target (for example, `scsidev@0.4.0`).

The following example represents typical output from the SJIRJC command:

```
Device: scsidev@0.4.0
Number of Drives: 1
Number Drive Pairs: 1
Number of Import/Export Elements: 0
Number of Import/Export Pairs: 1
Number of Slots: 7
Number of Slot Pairs: 1
Number of Transport Elements: 1
Number of Transport Pairs: 1
Initialize Element Status Supported
Auto Eject Supported
```

## tur

The `tur` program sends a TEST UNIT READY command to all SCSI devices attached to the system, or, if the optional `-a` *b.t.l* argument is specified, then the device at the specified ordinal SCSI address, where *b* is the logical SCSI bus, *t* is the SCSI target, and *l* is the SCSI lun on that target. The following example shows the format to use for the `tur` program:

```
tur [-a b.t.l]
```

# Data Management

This section provides a command line reference for Backup commands to use for data management. Many of these commands are also automatically invoked by the Backup server during scheduled backups. The commands for HSM and Archive are only available when you enable the optional modules for these features on the Backup server.

# savegrp

The `savegrp` program runs a group of Backup clients through the `save` process to back up filesystem data. The group of clients is selected by the name assigned (see “NSR Group” on page 279). Typically, `savegrp` is invoked automatically, as specified by each group’s `NSR group` resource.

If you do not specify a group name, the Backup group named Default is used. If you specify a group name, clients whose `nsr_client` resources specify the named group in their Group attribute are included. If you specify an explicit client list with the `-c` *client-name* option, the `savegrp` program only includes the named clients in the backup and ignores other members of the group.

If you enable the Clone attribute for the named group, the `savegrp` program automatically invokes a clone of the save sets backed up during the `save` session. The client save sets and their associated file indexes are cloned before the bootstrap save set is generated, which allows the bootstrap to track both the original save sets and their clones. The bootstrap save set is cloned as well. Cloned save sets are sent volumes assigned to the clone pool specified in the `NSR group` resource.

If a client’s Save Set attribute specifies “All,” the `savegrp` program requests a list of the filesystems to perform the `save` program on (this is called a *probe*). The probe expands “All” into a list by searching for local and automatically mounted filesystems on the client machine (NFS mount points and manually mounted filesystems are generally not included in the list gathered by the probe).

You cannot run more than one occurrence of the `savegrp` program on the *same* group at the same time; the program exits with an error message. If you run *different* groups at the same time, each group runs `save` program sessions up to the limit specified in the Parallelism attribute for the `nsr_client` resource (the default value for Parallelism is 4). However, the Backup server only allows `save` program sessions up to the limit specified in the server’s Parallelism attribute to write to one backup device at a time. Each save set generates a separate `save` program session, regardless of the client it originates from.

When the `save` process (and, if enabled, clone process) is complete, a notification with an Event value of “savegrp” and a Priority value of “notice” is sent to the `nsr_notification` system. This is generally set up to send e-mail to the root user to indicate the success or failure of the backup, the clients backed up during the `savegrp` execution, and the data saved.

The following example describes the format and options available for the `savegrp` program:

```
savegrp [see “Options”] [-R | -G] [group-name]
```

Options:

```
[-EIOmnpv] [-l level | -C schedule]
[- e expiration] [- t date] [-r retries]
[-P printer] [-W width] [-c client [-c client...]]
```

- Use the `-c` *client* option to run `savegrp` on a specific client or clients. When you specify this option, only the named clients from the specified *group-name* are run.
- Use the `-C` *schedule* option to specify the name of the `nsr_schedule` resource to use for the automatic save level selection process.
- Use the `-e` *expiration* option to specify the date when the saved data is to expire. If you use the special value of “forever” for expiration, the volume the data resides on never expires. This is typically used for migration or archive volumes. By default, no explicit expiration date is assigned.
- Use the `-E` option to estimate the amount of data that is generated by each save set before the save operation is performed. This option results in a double traversal of the filesystems: once to generate an estimate and again to perform the actual save operation. The data itself is only read from the disk on the final pass, because the estimate is performed by accessing the anode information.
- Use the `-G` option to run only the group, without restart semantics.
- Use the `-I` option to disable the save operation performed on each client’s file index.
- Use the `-l` *level* option to specify the level of the save.
- Use the `-m` option to disable monitor status reports, including all the nsr_notification actions.
- Use the `-n` option to cause `save` to perform an estimate as described for the `-E` option, but not to perform an actual save after it generates the estimate. The `-m` option is implied when you use the `-n` option.
- Use the `-O` option to only save each client’s file index. For the server, this results in a save of the bootstrap as well. By default, the Backup server’s bootstrap is backed up any time a group that it is a member of runs through a scheduled or manually invoked `savegrp` execution. The client file indexes and server bootstrap are a vital part of the disaster recovery procedure.
- Use the `-p` option to run the probe on each client. This provides information on the filesystems and level of `save` to perform on each client, without an actual save of the data. The `-m` option is implied when you use the `-p` option.
- Use the `-P` printer option to specify the printer that the `savegrp` program should send bootstrap information to upon completion of the backup.
- Use the `-r` *retries* option to specify the number of times the Backup server should retry failed clients before the `savegrp` program declares the client backup failed. The default value for this option is taken from the `NSR group` resource. Abandoned saves are not retried, because they may eventually be completed. A retry is not attempted if the `-p` option is specified.

- Use the `-R` option to use the information stored on the Backup server to restart a group that was previously terminated (generally, this is due to a crash of the Backup server during a backup).
- Use the `-v` option to run the `savegrp` program in verbose mode.
- Use the `-W` *width* option to format the `savegrp` output or notification messages. The default width is 80.

## save

The `save` program, which resides on each Backup client, saves files. You can monitor the progress of a `save` operation using the X Window System-based `nwadmin` program or the `curses`(3X)-based `nsrwatch` Administration program.

If you do not specify a path argument either on the command line or through the `-I` option, the current directory that `save` is invoked from is saved. The `save` program saves a directory by saving all the files and subdirectories it contains. The `save` program does not cross mount points, and it does not follow symbolic links. If you mount the paths indicated from a network file server, the `save` program instructs you to run the `save` program on the remote machine, or use the `-L` option.

Each file in the subdirectory structures specified by the path option is encapsulated in a Backup save stream. This stream of data is sent to a receiving process on the Backup server, which processes the data and adds entries to the client file index for each file in the stream. The data is then directed to long-term storage, either on the server or the designated storage node.

**Caution –** The server's bootstrap and the client file indexes are only backed up automatically during a scheduled or manual backup that invokes the `savegrp` program. If you *never* run the `savegrp` program, either a scheduled or manually invoked backup, you do not have the server bootstrap or client file indexes that are vital to the disaster recovery process.

The following example describes the format and options available for the `save` program:

```
save [-BEiLnqvx] [-s server] [-c client-name]
[- N name] [-e expiration] [-f directory-file]
[-b pool] [-F file] [-I input-file] [-g group]
[-l level] [-t date] [-m masquerade] [-W width]
[path...]
```

- Use the `-b` *pool* option to specify a particular destination pool for the save sets.

- Use the `-B` *option* to force a save of all connecting directory information, from the root (`/`) to the point of invocation.
- Use the `-c` *client-name* option to specify the client name that starts the save session. This is useful for clients with multiple network interfaces and, hence, multiple hostnames. You can use the option to create multiple client file indexes for the same physical client machine. This option does not specify the network interface to use; the network interface is specified in the Network Interface attribute of the `nsr_client` resource.
- Use the `-e` *expiration* option to set the date when the save set expires. When a save set has an explicit expiration date, the save set remains both browsable and nonrecyclable until it expires. After the expiration date, the save set is nonbrowsable. If it has expired and also passed its retention time, the save set becomes recyclable. By default, explicit save set expiration dates are not used.
- Use the `-E` option to estimate the amount of data that is generated by each save set before the save operation is actually performed. This option results in a double traversal of the filesystems: once to generate an estimate and again to perform the save operation. The data itself is only read from the disk on the final pass, because the estimate is performed by accessing the inode information.
- Use the `-f` *dirfile* option to specify the file from which the save program should read the prototype default directives. A *dirfile* value of "-" causes the default directives to be read from standard input.
- Use the `-F` *file* option to save only files whose change time is newer than the file modification date of the specified file.
- Use the `-g` *group* option to denote the group to save. Use this option to determine the specific pool to which save sets from the specified group should be written.
- Use the `-i` option to instruct the `save` command to ignore any `.nsr` directive files encountered in the subdirectory structures saved.
- Use the `-I` *input-file* option to read the paths to save from the named text file, in addition to the paths listed on the command line. The paths must be listed one per line. If no paths are listed on the command line, only the files contained in the paths listed in *input-file* are saved.
- Use the `-l` *level* option to specify the level of the save.
- Use the `-L` option to perform a save from the local Backup client, even when files are from a network fileserver. To recover files, you must run the `recover` program with the same `-c` *client* argument used to save the data.
- Use the `-LL` option to treat the backup as a local save and print an extra line at the end of the completion report in the form "complete savetime=*number*" where *number* is the savetime of the save set created by this backup. This option is meant for use by the `savegrp` command for automatic cloning.
- Use the `-m` *masquerade* option to specify the tag to precede the summary line in the savegroup completion report.
- Use the `-n` option to estimate the amount of data that will be saved, without performing a save operation.
- Use the `-N` option to specify the symbolic name of the save set. By default, the most common prefix of the path argument is used as the save set name.
- Use the `-q` option to run the `save` program in quiet mode. This option generates only summary information and error messages.

- Use the -t *date* option, in nsr_getdate(3) format, to specify the date after which files must have been modified to qualify for a save.
- Use the -v option to run the save program in verbose mode.
- Use the -W *width* option to format summary information output.
- Use the -x *option* to cross mount points during the save operation.

# savefs

The savefs program is used by the savegrp program to probe a client for its filesystems and recent save times. Running savefs directly to perform a save is not recommended. However, you can safely invoke savefs manually with the –p option to probe the client and produce a preview report of the save sets (and levels) that a savegrp will back up. When probing, *savefs* does not actually save data, but instead produces a machine-parsable report that describes the layout of the client's filesystems. The -p option provides command line access to the same information you obtain with the Group Control>Preview feature available in the GUI version of the Administration program.

If a filesystem argument is not provided with the savefs command line, the filesystems listed in the Save Set attribute are probed. If the save set list consists of the keyword "All," then the filesystem tables (/etc/vfstab on Solaris, /etc/mnttab on SCO, and a kernel table on AIX) are examined to determine which filesystems to save. Only local, mounted filesystems are considered by the probe.

Metadevices within the Sun Solaris Online DiskSuite and Logical Volumes within the HP-UX Logical Volume Manager are treated similar to independent disks. This approach allows each to be saved in its own session, assuming sufficient parallelism.

Care should be taken when the Clients resource explicitly lists the save sets, for two primary reasons. First, this list must be manually updated when new filesystems that need saving are added. Second, since savefs only stops at the end of a path or a mount point, if you list two save sets in the same filesystem and one is a subdirectory of the other, the subdirectory is saved twice.

You can specify filesystem arguments to limit the filesystem saves to only those specified, but the specified filesystems must appear on a Save Set list for this client (see the –F option).

The following example describes the format and options available for the savefs program:

```
savefs –p [options] [filesystem...]
[–M filesystem...]
```

The following lists the valid values for *options*:

```
[–BEFnpqRv] [–s server] [–N name] [–g group]
[–l level | –C schedule] [–e expiration]
[–f filename] [–W width] [–t date] [–T seconds]
```

- Use the `-B` option to force a save of all connecting directory information from root (“/”) down to the point of invocation. This option is used by `savegrp`, for example, when saving the server’s bootstrap information.
- Use the `-C` *schedule* option to specify the name of the schedule to use when automatically determining the save level. If this option is not specified, `savefs` uses the schedule named by the Clients resource for the specified filesystem.
- Use the `–e` *expiration* option to specify the expiration date for the saved data (in `nsr_getdate` format). By default, no explicit expiration date is used.
- Use the `–E` option to walk the filesystems specified and estimate the amount of data that the save will generate. Without this flag, the estimated size is zero. Note that this flag consumes an amount of time proportional to the number of files in each filesystem. This is because the entire directory is walked before any saving begins and walked again when actually saving the directory. The file data is only read from the disk the last time. In many cases, the overhead for using this flag is small and is well justified.
- Use the `-f` *filename* flag to specify the file from which application-specific modules (ASMs) should take their directives. By default, these are taken from the Directives resource named by the Directive attribute in the Clients resource for each client.
- Use the `-F` option to save every argument like a filesystem, even if the arguments are not listed in the filesystem tables or the Clients resource.
- Use the `-M` option, as part of a probe, to signify that all subsequent filesystems should be probed for their ability to be migrated. This option is quietly ignored on systems that do not support file migration.
- Use the `-g` *group* option to restrict the scope of the client to a particular group. If this option is not specified, save sets from all instances of the Clients resource for this client are used, regardless of the group. This value is also passed on to **save**, which uses it to select a specific media pool.
- Use the `-l` *level* option to specify the level of save to perform. There are 12 levels: **full**, levels **1** though **9**, **incr**, and **skip**. **Full** specifies that all files are to be saved. **Incr** specifies incremental saves in which only those files modified since the most recent save, at any level, are saved. **Skip** causes no files to be saved. Levels **1** through **9** save all files modified since any *lower* level save was performed. For example, if you did a `Full` on Monday, followed by a level `3` save on Tuesday, a subsequent level `3` save on Wednesday contains all files modified or added since the Monday `Full` save. If you do not specify a level, the save level is determined automatically from the Backup client’s schedule. Using the history of previous saves maintained by **nsrmmd** on the Backup server, `savefs` accurately computes

the time for the given level. When tapes are deleted, savefs uses media information on the server to automatically adjust the time computed for saves based on previous save levels.

- Use the `–n` option to have `savefs` accurately estimate the amount of data generated, as described for `–E`, but not actually save any data.
- Use the -N name option to assign the symbolic name for the save sets. By default, the first *filesystem* argument is used as the name.
- Use the `–p` option to list the name of the filesystems, the level of save that would be performed, and the file modification time of files to be saved, but not actually perform the save. This information is gleaned from an operating system-specific file and the Schedules resource.
- Use the `–q` option to run `savefs` in quite mode. Only summary information and error messages are displayed.
- Use the `-qq` option to run savefs in really quiet mode, and display only error messages.
- Use the `-R` option to cause `savefs` to echo a simple "succeeded" or "failed" message as it is completed. This option is automatically used by the `savegrp` program when it runs `savefs`.
- Use the `-s` *server* option to specify the Backup server for savefs to use.
- Use the `-t` *date* option to specify the date (in `nsr_getdate` format) for `savefs` to use as a base for calculating the level. If this option is not specified, the current time is used.
- Use the `-T` *seconds* option to specify the inactivity timeout, in seconds, for `savefs`. If `savefs` detects that the local server has not made progress in the specified time, it concludes that the `save` program is not responding. A message is printed to `stderr` and `savefs` exits normally. This option should only be used on Backup server machines.
- Use the `-v` option to run `savefs` in verbose mode. This option results in a lot of debug-style output. This option is automatically used by the `savegrp` program when it probes for the ability of the client's `savefs` to support multiple versions.
- Use the `-W` *width* option to specify the width used for formatting output or notification messages. The default value for *width* is 80.

## savepnpc

The `savepnpc` program, like the `save` program, saves files to long-term storage. Before performing a save operation, `savepnpc` performs any pre-processing commands that exist in the `/nsr/res/`*`group_name`*`.res` file. If the pre-processing command fails, `savepnpc` exits with an error code and `save` is not performed. At the end of a successful save of the last save set on the client, `savepnpc` performs any post-processing commands that exist in the `/nsr/res/`*`group_name`*`.res` file. An optional timeout condition may be set to indicate at which point in the post-processing commands must be run without waiting for the last save set to back up.

The Timeout attribute is set in the same /nsr/res/*group_name*.res file as the pre- and post-processing commands. All of the results from the savepnpc program are logged in the /nsr/res/savepnpc.log file.

The /nsr/res/*group_name*.res file is automatically created the first time you run a backup group with a client that has the savepnpc command entered in the Backup Command attribute of the Clients resource. The format looks similar to the following:

```
type: savepnpc;
precmd: /bin/true;
pstcmd: /bin/true, “/bin/sleep 5”;
timeout: “12:00pm”;
```

You can edit the Precmd field to contain any number of commands, separated by commas, to run prior to the start of the save operation on the client's first save set. You can also edit the Postcmd field to contain any number of commands, separated by commas, to run at the end of the save operation on the client's last save set or the timeout condition indicated in the Timeout field, whichever comes first. All fields in the file must terminate with a semicolon (;).

The command syntax for savepnpc is identical to the syntax described for “save” on page 327. If you create a customized script to enter in the client’s Backup Command attribute, the following rules apply:

- The savepnpc command must be part of the script.
- The filename of the script must begin with save or nsr, and cannot exceed 64 characters in length.
- The script must reside in the same directory as the save program (typically, /usr/bin).

# recover

The recover program searches (browses) the client file index for a specified client and recovers files from backup volumes to the specified client. The client file index entries are created when the files are backed up with the save command. When you use the interactive version of the recover program, nwrecover, the client file index is presented in a graphical display format that is similar to a UNIX filesystem.

In the automatic mode (-a option) or save set recover mode (-S option), the files specified on the command line are recovered immediately without browsing the client file index. Use of the save set recover mode (-S option) is restricted to users in the *operator* group. If you run the recover program without the -S option, Administrator and users in the *operator* group can recover any file.

You can specify one or more `path` arguments to limit the directories and files to just those you want to recover. If you specify the path argument, the beginning of each path name as it exists in the save set must exactly match one of the paths before it can be recovered. Filename matching using meta characters (for example, *, ?, or [...]) is not allowed. You can use a path that ends with a slash character to force a match to a specific directory.

The following example describes the format and options available for the `recover` program:

```
recover [-f] [-n] [-q] [-i {nNyYrR}]
[-d destination] [-c client] [-t date]
[-s server] [dir]
recover [-f] [-n] [-q] [-i {nNyYrR}]
[-d destination] [-c client] [-t date]
[-s server] -a path
recover [-f] [-n] [-q] [-i {nNyYrR}]
[-d destination] [-t date] -s server
-S ssid[/cloneid] [-S ssid[/cloneid]] [path]
```

- Use the `-a` option to cause the `recover` program to automatically recover files without browsing the client file index.
- Use the `-c` *client* to specify the name of the machine from which the save sets were originally saved. When you browse a directory that was saved by a different client, the pathnames displayed reflect the filesystem of the client that saved the files. By default, the `save` and `recover` programs determine the client machine name from the filesystem table. If you specified the `-L` option with the `save` program, the `-c` *client* option may not be necessary (see “`save`” on page 327 for information about the options available for the `save` program). You cannot use the `-c` *client* option in conjunction with the `-S` *ssid*[/*cloneid*] option.
- Use the `-d` *destination* option to specify the destination directory where you relocate the recovered file. Relative paths are interpreted in relation to the current working directory.
- Use the `-f` option to force recovered files to overwrite any existing files whenever a filename conflict occurs. This option is the equivalent of specifying the combined `-iY` option.
- Use the `-i` option with one of the following choices to specify the initial default overwrite response to use when a file name conflict occurs: `nNyYrR`. You can only specify one letter choice in conjunction with the `-i` option. The `-i` option produces the same results as the `uasm -i` option when you run `uasm` in recover mode.
- Use the `-n` option to use the `recover` program without creating any directories or files.
- Use the `-q` option to turn off the default verbose mode for the `recover` program.

- Use the `-s` *server* option to specify the Backup server from which you want to recover data. This option is required when you use the save set recover mode (`-S`). If you omit the `-s` *server* option, the default is the server of the first directory marked for recovery, if the server is a network file server as well as a Backup server. If the server is not a network file server or a Backup server, the current server or a machine with a logical name of `nsrhost` entered in the host table is considered.
- Use the `-S` *ssid*[/*cloneid*] option to use the `recover` program in save set recover mode. Use this mode to implement batch file recovery without the need for client file indexes. The value of ssid specifies the save set IDs for the save sets you want to recover. When multiple clone instances exist for a save set, you can specify a clone ID to select the particular clone instance you want to recover. If you do not specify the path argument, the entire contents of the save set are recovered.
- Use the `-t` *date* option to display or recover files as of the specified date. You cannot use this option in conjunction with the `-S` *ssid* option.

Refer to the `recover(1m)` man page for more information on how to use the `recover` program in interactive mode, as well as to view a listing of the more common error messages encountered.

# nsrmig

The `nsrmig` program migrates files to the volumes labeled for a Migration pool type. The migrated files are replaced with a stub (a symbolic link) that points to a copy of the file made during premigration with the `nsrpmig` program. If you access the stub later, the file is automatically recalled to disk from the migration volume by the Backup server or storage node.

The criteria for migration is defined in the Migration resource on the Backup server. Migration is usually an automatic process controlled by the Backup server. The criteria most often employed is last access time. Only regular files are premigrated and, ultimately, migrated.

If you do not specify a path argument, the current directory is migrated. The `nsrmig` program does not cross mount points, and it does not follow symbolic links.

The following example describes the format and options available for the `nsrmig` program:

```
nsrmig [-nvx] [-l percent] -s server]
[-t savetime] [-W width] [path]
```

- Use the -l *percent* option to specify a goal percentage for the nsrmig program to use. Migration stops when the goal percentage is reached. If the goal percentage is already reached before you invoke nsrmig, the program exits without performing any further migration. If you do not specify the -l option, the goal percentage is read from the appropriate migration client resource.
- Use the -n option to estimate the number of files and total size that are freed by replacing the files that qualify for migration with a stub, but do not replace the files with stubs.
- Use the -s *server* option to specify the machine to use as the Backup server. If you omit this option, the default machine considered is either the current machine (if it is a Backup server) or a machine with the logical name of nsrhost entered in the host table.
- Use the -t *savetime* option to migrate files that were premigrated at the specified *savetime*.
- Use the -v option to cause the save program invoked by nsrpmig to provide detailed information as it proceeds.
- Use the -W *width* option to specify the width that nsrmig should use to format summary information to standard output. The default width used is 80.
- Use the -x option to instruct nsrmig to cross mount points.

Refer to the nsrmig(1m) man page for further details and common error messages encountered.

## nsrpmig

The nsrpmig program premigrates files that are identified as candidates for migration, as defined in the Backup server's Migration resource. The premigration process invokes the save program to immediately make a copy of the specified file to a backup volume labeled for migration data. When the file is later migrated, the resident file is replaced with a marker that refers to the premigrated copy on volume. You can only premigrate regular files.

The nsrpmig program does not cross mount points or follow symbolic links. If you mount the path to be saved from a network file server, the nsrpmig program issues a message that instructs the user to run the save program on the remote machine or use the -L option with nsrpmig.

The nsrpmig program examines the directive files (.nsrhsm) encountered in each directory to determine any special instructions to apply when saving files (for example, compression and skip directives). The directive files ordinarily used by Backup for save and recover operations (.nsr) are ignored by the nsrpmig program.

The nsrpmig program is only available for use when an enabler code for the Backup HSM is present on the Backup server.

The following example describes the format and options available for the `nsrpmig` program:

```
nsrpmig [-BEiLnpqvx] [-s server] [-N name]
[- f dirfile] [-b pool] [-g group]
[-m masquerade] [-W width] [-C clone-pool]
[-I input-file] path
```

- Use the `-b` *pool* option to specify the volume pool to which the premigrated data should be saved. Migrated data must reside on separate volumes from either backed-up data or archived data. If you do not specify a pool, the Migration pool is selected by default.
- Use the `-B` option to force a save of all connecting directory information, from the root (/) to the point of invocation.
- Use the `-C` *clone-pool* option to generate a clone of the premigrated save set to the specified clone pool. Clones of migrated data must reside on separate volumes from either backed-up or archived clone data. If you do not specify a clone pool, the Migration Clone pool is selected by default.
- Use the **-E** option to instruct `nsrpmig to` estimate the amount of data that the `save` program generates, then perform the save operation. The estimate is generated from the inode information, so the data is only read once.
- Use the `-f` *dirfile* option to specify a file that `nsrpmig` should read prototype default directives from [refer to the `nsr(5)` man page for more information on the default directives]. A value of "-" for *dirfile* causes the default directives to be read from standard input.
- Use `-g` *group* option to denote the group name to which the save set should belong. The Backup server uses this option to select a specific media pool.
- Use the `-i` option to instruct `nsrpmig` to ignore any `.nsrhsm` directive files encountered during the premigration process.
- Use the `-I` *input-file* option to instruct `nsrpmig` to read the paths to save from the file specified as *input-file* in addition to those listed on the `nsrpmig` command line. List each path on a separate line in the file specified by *input-file*. If you do not also specify paths on the command line, only the paths specified in *input-file* are saved.
- Use the `-L` option to instruct `nsrpmig` to perform a local save from the Backup client, even if the files originate from a network fileserver. To recover files that have been locally premigrated, run the `recover` program with the `-c` *client* option, where the value for *client* is the machine name of the Backup client that performed the save operation.
- Use the `-LL` option to instruct `nsrpmig` to perform a local save and print an extra line at the end of the completion in the format "complete savetime=*number*" where *number* is the save time of the save set created. The `savegrp` program uses this option when you specify automatic cloning.
- Use the `-m` *masquerade* option to specify a tag to precede the savegroup summary notification line. The `savegrp` and `savefs` programs use this option to aid in savegroup summary notifications.

- Use the -n option to estimate the amount of data that is generated by the save without performing the save. This option is similar to the -E option, except that data is not saved to a volume after the estimate is completed.
- Use the -N *name* option to instruct nsrpmig to use the symbolic name of the save set. By default, the path argument is used as the save set name.
- Use the -p option to cause the save program invoked by nsrpmig to exit with a status value of 0. The server uses this option to determine whether a client is installed properly.
- Use the -q option to cause the save program invoked by nsrpmig to display only summary information and error messages.
- Use the -s *server* option to specify the machine to use as the Backup server. If you omit this option, the default machine considered is either the current machine (if it is a Backup server) or a machine with the logical name of nsrhost entered in the host table.
- Use the -v option to cause the save program invoked by nsrpmig to provide detailed information as it proceeds.
- Use the -W *width* option to specify the width that nsrpmig should use to format summary information to standard output. The default width used is 80.
- Use the -x option to instruct nsrpmig to cross mount points.

See "save" on page 327 and "savegrp" on page 324 for more information on the save and savegrp program options described in this section. Refer to the nsrpmig(1m) man page for further details and common error messages encountered.

# nsrhsmck

The nsrhsmck program checks and corrects the consistency between the file stubs and the client file index entries for files migrated by HSM. The nsrhsmck program handles four situations:

- The first situation occurs when you rename the stub for a migrated file. In this situation, the stub with the original filename no longer exists. The nsrhsmck program corrects this situation by updating the client file index entry to reflect the new name given to the stub.
- The second situation occurs when you create a symbolic link that points to the same name in the Backup Instruction Buffer (IB) namespace as another symbolic link. The nsrhsmck program corrects this situation by replacing the duplicate with a symbolic link that points to the original symbolic link, rather than pointing directly to the Backup IB namespace.
- The third situation occurs when you delete the stub that points to a migrated file. This is known as the possible delete case. The term "possible" implies that the stub may reappear later, for example, if the stub is recovered using Backup. The nsrhsmck program corrects this situation by marking the index entry for the

migrated file as a possible deletion after 60 days. Note that if a file marked as possibly deleted is detected on disk before the index entry is later deleted, the index entry is unmarked as a possible deletion.

- The fourth situation handled by `nsrhsmck` occurs when an index entry that is marked as a possible deletion that has passed the 60 day expiration time. The `nsrhsmck` program corrects this situation by removing the expired entries from the HSM file index. Before it deletes an entry from the HSM file index, `nsrhsmck` makes a final check to make sure the file does not exist on disk.

You must specify a path on the command-line when you run `nsrhsmck`. Only files and index entries that fall under the path specified are examined for consistency.

The following example describes the options available for the `nsrhsmck` program:

- Use the `-c` option to instruct the `nsrhsmck` program to walk the HSM file index and delete entries marked as possibly deleted that have passed the 60-day expiration period.
- Use the `-d` option to instruct the `nsrhsmck` program to walk the HSM file index and mark any possible deletions that are detected.
- Use the `-f` option to instruct the `nsrhsmck` program to walk the filesystem on disk and search for duplicated links and renamed stubs.
- Use the `-M` option to tell the `nsrhsmck` program that it is being run in master mode by `nsrexecd` or another Backup daemon, and, therefore, to log messages with timestamps as well as perform other behavior expected by `nsrexecd`. This option is not advised for manual operation; it is used by the Backup server when `nsrhsmck` is automatically invoked.
- Use the `-n` option to instruct the `nsrhsmck` program to report on any inconsistencies found, without correcting them.
- Use the `-s` *server* option to specify the machine to use as the Backup server. If you omit this option, the default machine considered is either the current machine (if it is a Backup server) or a machine with the logical name of `nsrhost` entered in the host table.
- Use the `-v` option to run `nsrhsmck` in verbose mode. You can specify this flag up to three times on the command line to achieve the highest level of verbosity. Note that the verbose mode can produce an extremely large quantity of output and is not recommended for use in most situations.

## nsrarchive

The `nsrarchive` program archives files, including directories or entire filesystems, to the Backup server. You can use the `nwadmin` or `nsrwatch` programs to monitor the progress of an archive operation. Only users on the Administrator and Archive Users lists have the required privileges to run the `nsrarchive` program. Additionally, you can allow or disallow public archives through an option in the NSR (or Server) resource, which enables other clients to recover data archived from a particular client machine.

If you do not specify a path argument, the current directory is archived. The `nsrarchive` program archives all the files and subdirectories contained in a directory, but does not cross mount points or follow symbolic links. If the paths to be archived are mounted from a network fileserver, the `nsrarchive` program returns a message that instructs you to run the `nsrarchive` program on the remote machine or use the `-L` option.

The `.nsr` directive files encountered in each directory are read by default. The directive files contain instructions on how specific files should be archived (for example, compression).

Each file in the subdirectory structures specified by the path option is encapsulated in a Backup save stream. This stream of data is sent to a receiving process on the Backup server, which processes the data and adds entries to the client file index for each file in the stream. The data is then directed to long-term storage, either on the server or the designated storage node.

The following example describes the format and options available for the `nsrarchive` program:

```
nsrarchive [-BiLnpqvxVy] [-b pool] [-C clone-pool]
[-f filename] [-G remove] [-N name] [-R name]
[-s server] [-T annotation] [-W width] [path...]
```

- Use the `-b` *pool* option to specify a destination pool for the archive save sets. This option overrides the automatic pool selection typically used by the server. Archive data must be directed to volumes specifically labeled for a pool type of Archive. If you do not specify a pool, the Archive pool is selected by default.
- Use the `-B` option to force an archive of all the connecting directory information, from root (`/`) to the point of invocation.
- Use the `-C` *clone-pool* option to automatically generate a clone of the archived save sets to the specified clone pool. Cloned archive data must be directed to volumes specifically labeled for a pool type of Archive Clone. If you do not specify a clone pool, the Archive Clone pool is selected by default.
- Use the `-E` option to estimate the amount of data that the archive generates, followed by the archive. Note that the estimate is generated from the inode information; therefore, the data is only read once.
- Use the `-f` *filename* option to specify a file from which `nsrarchive` should read the default directives to apply to the archive data (refer to the `nsr(5)` man page for further information on directives). A value of "-" for *filename* causes the default directives to be read from standard input.
- Use the `-G remove` option to groom (remove) files after they are successfully archived. If you specify cloning or verification options as well, the groom operation is not performed until those operations are completed successfully. Unless you also specify the `-y` option, you are prompted for removal of top-level directories. The `nsrarchive` program creates a temporary file that contains a list

of all the files and directories to be groomed. The temporary file is placed in the directory specified by the TMPDIR environment variable, or in the /tmp directory if the environment variable is not defined.

- Use the -i option to instruct the nsrarchive program to ignore any directive files encountered in the subdirectories that are archived.
- Use the -L option to perform a local archive from the Backup client, even when the files are from a network file server.
- Use the -n option to estimate the amount of data that is generated by the archive without performing the actual archive. This option is similar to the -E option, except that data is not saved to a volume after the estimate is completed.
- Use the -N *name* option to instruct nsrarchive to use the symbolic name of the archive save set. By default, the first path argument is used as the value for *name*.
- Use the -p option to instruct nsrarchive to exit with a status of 0. This Backup server uses this option to determine whether the client is properly installed.
- Use the -q option to cause nsrarchive to run in quiet mode and display only summary information and error messages.
- The -R *name* option should only be used by the nsralist program, which handles the execution of the archive requests. Updates to the named archive request resource occur when the Backup server specifies this option.
- Use the -s *server* option to specify the machine to use as the Backup server. If you omit this option, the default machine considered is either the current machine (if it is a Backup server) or a machine with the logical name of nsrhost entered in the host table.
- Use the -T *annotation* option to assign an arbitrary text string of 1024 characters or fewer to the archive save set. The string specified as *annotation* is used by the nsrretrieve program to browse the media database for archive save set entries to retrieve back to local disk. The annotation is a mandatory requirement for all archive save sets; if you omit this option, you are prompted for it before the process continues.
- Use the -v option to cause nsrarchive to run in verbose mode.
- Use the -V option to verify each archive save set.
- Use the -W *width* option to specify the width that nsrarchive should use to format summary information to standard output. The default width used is 80.
- Use the -x option to instruct nsrarchive to cross mount points.
- Use the -y option to automatically enter an affirmative response to any queries generated by the nsrarchive program.

## nsrretrieve

The nsrretrieve program is used to restore archived save sets from the archive volumes managed by the Backup server or storage node. You do not browse client file index entries for archived save sets as you do for regular save sets; you search for a specific annotation string to identify the archive save set you want to retrieve.

The use of `nsrretrieve` is restricted to users on the Backup server's Administrators and Archive Users list. If the `nsrretrieve` program is not run by root or a user defined in the *operator* group, or the Public Archives attribute of the Server resource is not enabled, only the owner of the archived files can retrieve them.

The following example describes the format and options available for the `nsrretrieve` program:

```
nsrretrieve [-f] [-n] [-q] [-i {nNyYrR}] [-d destination] -s server [-
S ssid[/cloneid]]... [-A annotation]...
[path]...
```

- Use the `-A` *annotation* option to specify the archive save set to retrieve. An annotation is a regular expression that uniquely identifies a single archive save set. The regular expression is of the form used by the `grep(1)` command.
- Use the `-d` *destination* option to specify the destination directory where you want to relocate the retrieved files.
- Use the `-f` option to force retrieved files to overwrite any existing files whenever a filename conflict occurs. This option is the equivalent of specifying the combined `-iY` option.
- Use the `-i` option with one of the following choices to specify the initial default overwrite response to use when a filename conflict occurs: `nNyYrR`. You can only specify one letter choice in conjunction with the `-i` option. The `-i` option produces the same results as the `uasm -i` option when you run `uasm` in recover mode. Refer to the `usam(1m)` man page for a detailed explanation of how to use the `uasm -i` option.
- Use the `-n` option to use the `nsrretrieve` program without actually creating any directories or files.
- Use the `-q` option to cause `nsrretrieve` to run in quiet mode and display only summary information and error messages.
- Use the `-s` *server* option to specify the machine to use as the Backup server. If you omit this option, the default machine considered is either the current machine (if it is a Backup server) or a machine with the logical name of `nsrhost` entered in the host table.
- Use the `-S` *ssid*[/*cloneid*] option to specify the ssid for the save set to retrieve. If multiple clone instances exist for an archive save set, you can also specify the clone ID, to select the clone instance that you want to retrieve the data from. If you do not specify a path argument, the entire contents of the archive save set are retrieved. To restrict the retrieval to particular directories or files that match a given path prefix, specify the exact pathname.

# nsrclone

The nsrclone program makes new copies of existing save sets. The operation is automatic when you enable the Clones attribute of a NSR group resource. You can also run nsrclone on a manual basis from the command line.

Although the command line options enable you to specify a volume name or identifier, nsrclone always copies complete save sets, regardless of how many volumes the save set components reside on. The nsrclone program does not copy volumes; instead, it copies the original save sets specified from one volume to a volume assigned to a special pool for clones. If the first destination volume cannot hold all the save sets to be cloned, another volume from the same clone pool is chosen.

If you use the -c and -N options together, nsrclone creates a super-full copy for the given client save set. The super-full copy is a feature that is supported only under HSM. It automatically creates a clone of the most recent complete full backup of the named client and save set, along with any HSM migration save sets referred to by the full backup. Super-full copies should be cloned to a volume from a migration clone pool. If no migration save sets are referenced by the most recent full backup, only the full set is cloned.

The nsrclone program, in cooperation with the nsrmmd daemon, guarantees that each save set has only one clone on a given volume. When you specify a volume name or identifier, the copy of the save sets on that volume are used as the source. When you specify save sets explicitly, those with existing multiple copies are automatically chosen; copies of save sets that exist on volumes in an autochanger or silo are chosen over those that require operator intervention. You can also specify which copy of a save set to use as the source, with the -S option.

The following example describes the format and options available for the nsrclone program:

```
nsrclone [-v] [-s server] [-b pool]
{-f file | volume-name}
nsrclone [-v] [-s server] [-b pool] -S
{-f file | ssid}
nsrclone [-v] [-s server] [-b pool] -V
{-f file | volumeid}
nsrclone [-v] [-s server] [-b pool]
-c client -N saveset
```

- Use the -b *pool* option to specify the name of the clone pool to which the data should be migrated. If you omit this option, the cloned save sets are automatically sent to the Default Clone pool.
- Use the -c *client* option, in conjunction with the -N option, to specify a client whose save sets should be considered for a super-full copy.

- Use the `-f` *file* option to instruct `nsrclone` to read the volume names, volume identifiers, or ssids from the text file specified.
- Use the `-s` *server* option to specify a Backup server to migrate save sets from. If you omit this option, the current machine is selected by default.
- Use the `-S` option to specify one or more specific ssids. You can issue the `mminfo -v` command to determine the value to use for *ssid* (see "`mminfo`" on page 348 for details).
- Use the `-v` option to run `nsrclone` in verbose mode. This mode provides additional information during the process, for example, messages about save sets that cross volumes.
- Use the `-V` *volid* option to specify the name of the volume.

Refer to the `nsrclone(1m)` man page for examples and error messages for the `nsrclone` program.

# nsrssc

`nsrssc` consolidates the most recent level 1 (partial) save set and its corresponding full level save set into a new full level save set. This consolidation process effectively achieves the same outcome as a full level backup when the partial backup is completed.

Normally, `nsrssc` is invoked with `savegrp` (8) as part of a consolidation backup. During the consolidation level backup, `savegrp` automatically generates a level 1 backup; then it calls `nsrssc` to create a consolidated backup using the latest full level save set.

Compared to `savegrp` (8) method, using nsrssc directly gives the user greater control on how save set consolidation is performed. For example, savegrp (8) method always reverts to a full backup whenever save set consolidation fails. If the user does not want to revert to a full level backup when the process fails, then the user manually does a level 1 backup and consolidates the data using nsrssc.

Using `nsrssc` allows for greater flexibility in scheduling backups and save set consolidation. With `nsrssc`, the level 1 backup can be scheduled to take place at a different time than save set consolidation.

If `nsrssc` is executed manually, the most recently -backed up save set must be a level 1 save set; otherwise, the consolidation will not be successful.

The `nsrssc` requires at least two active devices. The consolidation process uses simultaneous device reads and writes to create its consolidated save set. This mechanism also creates a restriction upon the location of the newly created save set. The new saveset cannot be created on the same volume on which the partial or full save set from which it was derived reside.

## Performing Save Set Consolidaton With `nsrssc`

The following examples demonstrate how save set consolidation can be performed. In both examples, a save set defined in a group named `lab` is consolidated for client `popeye`.

Example 1:

To complete a save set consolidation, perform the following commands:

1. **`# \210savegrp \-G lab \-l 1 \-I\fP\s0`**
2. **`# \s10nsrssc \-c mars \-N/etc\fP\s0`**

In this example, the procedure is almost the same as completing the following command:

```
# \-G lab \-l c
```

The differences are:
- no index and bootstrap is backed up after data is consolidated;
- if a failure occurs during the consolidation process, a full backup is not performed

Example 2:

To direct level 1 data to a disk cache (a file-type device) and have the level 1 save set removed after a full level save set is consolidated, perform the following commands:

1. **`# \s10savegrp \-G lab \-l 1 \-I\fp\s0`**
2. **`# \s10nsrssc \ -c popeye \-N/etc \-r\fP\s0`**

This process removes the level 1; essentially, this process achieves the same outcome as a regular full backup.

Upon successful completion, **nsrssc** returns zero; otherwise, a non-zero value is returned.

Other error messages that might appear are:
- “You are not authorized to run this command”: only root or Backup administrators can run `nsrssc`.
- “Cannot contact media database”: `nsrmmd` is unavailable to answer queries or an additional Backup daemon might be finished. In this situation, the system administrator needs to determine if the Backup services need to be restarted.

There might be a short interval during startup when Backup services might be unavailable to answer any queries.

The following example describes the format and options available for the nsrssc program:

```
nsrssc -c client -N saveset [-p pool] [-vq]
```

The following command flag options are available:

- Use the -c option to run the name of the client whose save set should be included for the consolidation process
- Use the -N option to name the generated consolidated save set.
- Use the -p option to designate a pool to which the generated consolidated save set is directed. The pool may be any pool correctly registered with the nsrd(8)**.** It must be the same pool type as the previous full level save set. Possible pool values can be viewed by selected in the Pools menu from the Administration menu of Backup.

The Pool values are also listed in the **NSR** pools resource (see “NSR Pool” on page 290). If this option is omitted, then the consolidated save sets are automatically built on the volume(s) in which the media pool is the same as that of the previous full level save set.

- Use the -r option to remove the level 1 save set. If the level 1 save set is on tape, then the save set is expired. If the level 1 save set is on diskfile type volume, then the entire save set (both index entries and save set data on disk) are removed.
- Use the -v option to run nsrssc in verbose mode. This mode provides additional information during the process.
- Use the -q to only display error messages or messages of significance.

Refer to the nsrssc(8) man page for further information. Refer to the respective man pages for further information about the following commands:

- nsr_schedule(5)
- nsr_mminfo(8)
- savegrp(8)

# nsrstage

The nsrstage program is used on a manual basis to migrate existing save sets from one volume to another. The process begins with a clone of the specific save sets to the new volume specified, followed by deletion of the save set entries from the media database, and finally a removal of the save sets from the original source volume, if possible. The media database entries and save sets are not removed if the clone to the new volume does not succeed.

You can migrate save sets onto volumes that belong to any of the media types supported by Backup (for example, save sets on a file volume may be migrated to an optical disk). However, all volumes used as the destination of a `nsrstage` operation must belong to a Clone pool type. Refer to the `nsr_pool(1m)` man page for a description of the various pool types.

The `nsrstage` program does not perform simple volume migration; it migrates complete save sets. You can specify the copy (clone) of a save set to use as the migration source with the `-S` *ssid* option.

The following example describes the format and options available for the `nsrstage` program:

```
nsrstage [-v] [-s server] [-b pool] -m
[-S {ssid/cloneid}]
nsrstage [-v] [-s server] -C -V volume
```

- Use the `-b` *pool* option to specify the name of the clone pool to which the data should be migrated. If you omit this option, the cloned save sets are automatically sent to the Default Clone pool.
- Use the `-C` option to instruct `nsrstage` to perform a volume cleaning operation after the save sets have been migrated and their associated entries removed from the media database. You can only use this option with entries that are migrated from a file volume.
- Use the `-m` option to perform the actual migration.
- Use the `-s` *server* option to specify a Backup server to migrate save sets from. If you omit this option, the current machine is selected by default.
- Use the `-S` *ssid* (or *ssid*/*cloneid*) option to specify one or more specific ssids and clone IDs that you want to migrate. The *ssid* option is useful when you want to migrate individual save sets from a volume. The *cloneid* option is useful when you want to specify a particular copy of a save set for migration. The value of either identifier is an unsigned integer; when you specify both you must separate them with a slash (/) character. You can issue the `mminfo -v` command to determine the value to use for *ssid* or *cloneid* (see “`mminfo`” on page 348 for details).
- Use the `-v` option to run `nsrstage` in verbose mode. This mode provides additional information during the process, for example, messages about save sets that cross volumes.
- Use the `-V` *volume* option to specify the name of the volume that `nsrstage` should clean. You cannot use this option in conjunction with the `-S` or `-m` options.

Refer to the `nsrstage(1m)` man page for examples and error messages for the nsrstage program.

# scanner

The `scanner` program directly reads Backup media (such as backup tapes, optical disks, or files) to confirm the contents of a volume, to extract a save set from a volume, or to rebuild the Backup online indexes. You can only run this command as root. You must specify a device, which is usually one of the device names used by the Backup server. If the device is a tape drive, it must be a nonrewinding type.

If you invoke the scanner program without options (or only the `-v` option), the volume on the specified device is scanned and a table of contents is generated. The table of contents contains information about each save set found on the volume. By default, one line of information is written to standard output for each save set found on the volume. The information provides the client name, save set name, save time, level, size, files, ssid, and flag.

The following example describes the format and options available for the `scanner` program:

```
scanner [-Bimnpqv] [-s server] [-S ssid]
[-c client] [-N name] [-f file] [-r record]
[-t type] [-b pool] device [-x command argument-list]
```

- Use the `-b` *pool* option to specify the pool to which the volume should belong. This option only applies to volumes backed up by versions of Backup that did not store pool information on the media.
- Use the `-B` option, without the `-S` option, to quickly scan the tape to the location of the start of the bootstrap save sets. When the entire tape has been scanned, the ssid and tape file location of the most recent bootstrap save set is printed to standard output.
- Use the `-c` *client* option to instruct `scanner` to only process save sets that came from the machine specified by *client*. You can specify more than one client name in the same command line. You can also use the `-c` option in conjunction with the `-N` option, but only if you also specify the `-i` or `-x` option.
- Use the `-f` *file* option to start the scan at a specific media file number. See "`mminfo`" on page 348 for information on how to determine the media file number.
- Use the `-i` option to instruct `scanner` to rebuild both the client file indexes and media database from the volumes that are read. If you specify a single save set with the `-S` *ssid* option, only the entries from the specified save set are made to the client file index.
- Use the `-m` option to instruct `scanner` to rebuild only the media database for the volumes that are read.
- Use the `-n` option to run `scanner` without rebuilding the client file indexes or media database. This option provides a way to check the media without modifying the client file indexes or media database.

- Use the `-N` *name* option to process only save sets that match the specified name. The value of *name* should be a literal string. You can specify multiple names when you use this option in conjunction with the `-c` *client* option, but only if you also specify the `-i` or `-x` option.
- Use the `-p` option to print out information about each save set as it is processed.
- Use the `-q` option to display only error messages or messages of significance.
- Use the `-r` *record* option to start the scan at a specific media record number, to avoid a scan of potentially unused information. See "`mminfo`" on page 348 for information on how to determine the media record number.
- Use the `-s` *server* option when you run the `scanner` program on a storage node, to specify the controlling Backup server.
- Use the `-S` *ssid* option to extract the save set specified by *ssid*. When you use this option in conjunction with the `-i` or `-x` options, you can specify multiple *ssid* values. The save sets selected are in addition to any selected by the use of the `-c` and `-N` options. If you also specify the `-B` option, the value of *ssid* is assumed to be that of the bootstrap save set; only one *ssid* can be specified in this case.
- Use the `-x` *command* option, with an optional list of command arguments, to specify a UNIX command to execute on each save set scanned. This option can only be specified once per `scanner` command line, after the device specification.

Refer to the `scanner(1m)` man page for examples of `scanner` command usage and a list of common error messages.

# File Index and Media Database Management

The Backup client file indexes contain entries that enable users to browse and recover any files backed up by Backup that have not exceeded their assigned browse policy. The Backup media database managed by the server contains information about where the backed-up data resides. You can query the Backup client file indexes as well as the server's media database to obtain information.

## mminfo

The `mminfo` program reports information about Backup media and save sets. The default `mminfo` report displays information about the save sets that completed properly during the last twenty four hours. This report includes; the volume name, client name, creation date, amount of data saved to the volume, level of backup performed and the name of the save set.

See “Examples of `mminfo` Report Commands” on page 352 for a list of examples of how to use the `mminfo` command.

The following example describes the format and options available for the `mminfo` command:

```
mminfo [-avV] [-o order] [-s server] [report] [query] [volname...]
<report>: [-m | -B | -S | -X | -r reportspecification]
<query>: [-c client] [-N name] [-t time] [-q query specification]
```

- Use the `-a` option to apply the query to all complete, browsable save sets, not just those in the last 24 hours. This option is implied by the `-c`, `-N`, `-q`, `-m`, and `-o` options. When combined with a media-only report (`-m` or a custom report showing only media information), the `-a` option applies to all volumes, not just those with complete and browsable save sets.
- Use the `-B` option to produce a list of the bootstraps generated in the previous five weeks. The bootstrap report format is used, with one line of output printed for each matched save set. Each line shows the save date and time, save level, ssid, starting file number, starting record number, and volume.
- Use the `-c` *client* option to restrict the report information to the media and save sets that pertain to the specified client.
- Use the `-m` option to display a media-only report. This report displays information about each volume contained within the specified Backup server’s media database.

Use the `–v` option concurrently with the `–m` option to display; the internal volume identifier (volid), the number of the next file to be written and the media type.

Use the `–V` option concurrently with the `–m` option to display volume characteristics where:

  - The `d` flag, indicates that the volume is currently being written to.
  - The `r` flag, indicates that the volume is marked as read-only.

- Use the `-N` *saveset-name* option to restrict the reported information to the media and save sets pertaining to the specified save set name.
- Use the `-o` *order* option to sort the output in a specified order. *order* may be any combination of the letters `celmontR`, where:
  - `c`, client
  - `e`, expiration date of the volume
  - `l`, length or percentage of space used on the volume
  - `m`, media
  - `n`, saveset name
  - `o`, filename and record number
  - `R`, reverse

- `t`, the last time the media was accessed

The default sorting order for a saveset report is `mocntl`.

- Use the `-q` *queryspecification* option to add the given query constraint to the list of constraints on the current query. Multiple `–q` options may be specified, and combined with the shorthand query constraints `–c`, `–N` and `–t`. The syntax of the *queryspecification* is:

[!] *name* [*comp value*] [ , ... ]

`name`, is the name of a database attribute, such as “name=”hot”name=”hot”” Save Set” `comp`**,** is a valid comparator for the attribute, from the set “>”, “>=”,”=” `value`, is the value being compared.

The comparator and value must be specified for all attributes, except flags. If a string contains commas, quote the value using single or double quotes. The following is a valid string comparison:

```
name="Daily, ""hot"" Save Set"
```

Except for multiple character string values, all of the specified constraints must match a given save set and/or media volume. Numeric constraints can be specified by a, and all character string constraints can be specified by multiple possible values. For example,

```
%used>20,%used<80
client=mars,client=saturn
```

Refer to the CUSTOM QUERIES AND REPORTS section in the `mminfo(1m)` man page for further information on the syntax to use for the query specification.

- Use the `-r` *reportspecification* option to specify how a report is displayed. Specify the media and save set attributes to be displayed, order of the columns, column widths, and line breaks. The syntax of a *reportspecification* is:

*name* [(*width*)] [, *name* [(*width*)]...]

`name`, is the name of a database attribute

`width`, specifies how wide the column should be

- Use the `-s` *server* option to display volume and save set information from the specified Backup server. The default value for *server* is the current system.
- Use the `-t` *time* option to restrict the reported information to the media and/or save sets pertaining to the save sets created on or after *time*. Refer to the `nsr_getdate(3)` man page for a description of the recognized time formats. The default value for *time* is “yesterday.”
- Use the `-v` option to enable verbose display reports that include:

  - aborted completed purged and incomplete save sets
  - creation time
  - internal save set identifier (ssid),
  - An indicator of which portion of a save set resides on a volume.

  `c`, the entire saveset is contained on this volume.

  `h`, the head of the saveset is contained on this volume.

  `m`, a middle section of the saveset is contained on this volume.

  `t`, a tail section of a spanning save set is contained on this volume.

  - status of a save set, as indicated by:

  `b`, the save set is browsable with the `recover` command.

  `r`, the save set is recoverable with the `scanner` command.

  `E`, the save set has been marked eligible for recycling and may be over-written at any time.

  `S`, the save set was scanned, or rolled in. Rolled in save sets are not subject to the standard index management procedures and will remain in the file index until the user manually purges the save set.

  `a`, the save was aborted before completion. Aborted save sets are removed from the on-line file index by `nsrck`.

  `i`, the save is still in progress.

- Use the `-S` option to display a long, multi-line save set report for debugging. Each attribute of a save set is displayed in one of the following formats:

```
name=value
client:name
```

The first line of each multi-line group starts on the left margin and includes the save set identifier (ssid), save time, client and save set names. Subsequent lines for this save set are indented. The next line displays the level, the save set flags, the save set size, the number of files within the save set, and the save set expiration date. Extended attributes, clones and instances of the save set are displayed on the lines that follow.

- Use the `-V` option to display a more verbose report than that obtained through the use of the `-v` option.

The first line includes:

  - the size of each portion of a save set contained on this volume.
  - the creation date and time

The second line contains the following information:

- the save time in seconds since 00:00:00 GMT, Jan 1, 1970
- the internal save set identifier (ssid)
- the offset of the first and last bytes of the save set contained within section
- the media file number
- the first record within the media file containing data for this save set
- the internal volume identifier (volid)
- the total size of the save set
- the flag, indicating which part of the save set is contained in this media file (c, h, m, or t)
- save set’s status (b, r, a, or i).

- Use the `-X` option to prepare a save set summary report. This summary report breaks the save sets down into several overlapping categories:
  - the number of each level backup by type, performed on a save set.
  - the number of archived, migrated, empty and purged save sets.
  - the number of index save sets.
  - the number of incomplete save sets.

For recent usage, weekly and monthly summaries displayed, including the following information:

- the number of files saved in the time interval specified
- the number of save sets
- the total size, and average size per save set
- the average size per file
- the percentage of the amount saved for incrementals v.s. fulls

## Examples of `mminfo` Report Commands

The following examples provide a guideline for you to follow when you create your own customized queries. Shortened syntax, wherever acceptable, is shown.

To display all the information about all the volumes managed by the server:

```
# mminfo -m
```

To display media information from volumes that are labeled `mars.001` and `mars.002`:

```
# mminfo -m mars.001 mars.002
```

To display all save sets found in the file indexes named `/usr`:

```
# mminfo -N /usr
```

To display save sets named `/usr`, generated by a client named `venus`, backed up in the past week:

```
# mminfo -N /usr -c venus
```

To display save sets named `/usr`, generated by a client named `venus`, on a volume that is labeled `mars.001`:

```
# mminfo -N /usr -c venus mars.001
```

To display a media report of all volumes written on in the past week:

```
# mminfo -m -t `last week'
```

To display a media report of all non-full volumes, showing the percent used, pool name, and location of each volume:

```
# mminfo -a -r `volume,%used,pool,location' -q `!full'
```

To display a media report similar to the `-m` report that shows the barcode instead of the volume label:

```
# mminfo -a -r \ `state,barcode,written,%used,read,space,volexp'\
-r`mounts(5),space(2),capacity'
```

To display a verbose list of the instances of all save sets with more than one copy, sorted by save time and client name:

```
# mminfo -otc -v -q `copies>1'
```

To display all archive save sets with an annotation of “my project” for the past four months:

```
# mminfo -q'annotation=my project' \
-r”volume,client,savetime,sumsize,ssid,name,annotation" \
-t'four months ago'
```

## mmlocate

The `mmlocate` program accesses and manages the volume location information contained in the media database. Any user can use this command with the `-l` (default) or `-L` options. The `-c`, `-d` and `-u` options are limited to Backup administrators. Running `mmlocate` without any arguments lists all volumes and their locations for the specified server. (If you do not specify a server, the current host is used.)

If you use the `nsrjb` command to move a volume inside a jukebox, the location of a volume is set to the name of the jukebox.

The following example describes the format and options available for the `mmlocate` program:

```
mmlocate [-s server] [-l] [-n volume-name | -i volumeID | location]
mmlocate [-s server] -L
mmlocate [-s server] -d location
mmlocate [-s server] -c {-n volume-name | -i volumeID}
mmlocate [-s server] -u
{-n volume-name | -i volumeID} location
```

- Use the `-a` option to apply the query to all complete, browsable save sets, not just those in the last 24 hours. This option is implied by the `-c`, `-N`, `-q`, `-m`, and `-o` options. When combined with a media-only report (`-m` or a custom report showing only media information), the `-a` option applies to all volumes, not just those with complete and browsable save sets.
- Use the `-c` option to clear the location field for the specified volume.
- Use the `-d` *location* option to delete all volumes that show the given location. You receive a confirmation prompt prior to the deletion of each volume.

- Use the `-i` *volid* option to restrict the `mmlocate` operation to the specified volume ID.
- Use the `-l` *query* option to perform a database query using the supplied volume name, volume ID, or location. If you list the `-l` option without specific query requests, volumes without a set location are displayed.
- Use the `-L` option to list all locations found in the database.
- Use the `-n` *volname* option to restrict the operation to the volume name listed.
- Use the `-s` *server* option to access the server’s media database.
- Use the `-u` option to update the location for a volume. Locations are limited to a maximum length of 64 characters. You must also specify the `-n` *volname* or `-i` *volid* options and specify a location.

# mmpool

The `mmpool` program accesses pool information stored in the Backup server’s media database. You can also use the command to delete all the volumes in a particular pool. If you specify one or more volume names with the `mmpool` program, the report shows the pool to which each named volume belongs. By default, all volumes and their pools are displayed.

You cannot change the pool to which a volume belongs without relabeling the volume, which destroys all data stored on the volume. Pools are configured through a Backup administration tool, such as `nwadmin` or `nsradmin`. Use the administration tool to create and modify unique pools (see “NSR Pool” on page 290).

The following examples describe the format and options available for the `mmpool` program:

```
mmpool [-s server] [volume...]
mmpool [-s server] -d pool-name
mmpool [-s server] -l [pool-name]
mmpool [-s server] -L
```

- Use the `-d` *pool-name* option to delete all volumes for the given pool. You are prompted for deletion of each volume.
- Use the `-l` *pool-name* option to list all volumes and the pools to which they belong. If you specify a pool, `mmpool` only lists the volumes in that pool.
- Use the `-L` option to list the names of all of the pool resources configured on the server.
- Use the `-s` *server* option to specify the Backup server to act on. Refer to the `nsr(1m)` man page for a description of server selection.

## mmrecov

The mmrecov program recovers a Backup server's online file index and media database from backup volumes when either of the files is lost or damaged. Note that this command overwrites the server's existing online file index and media database. The mmrecov program is not used to recover Backup clients' client file indexes; you can use normal recover procedures for this purpose.

You must fully install and correctly configure the Backup server software and run a backup that includes the server's file index and media database before using the mmrecov program for the first time. If any of the Backup software is lost, reinstall the software from the distribution files before you run mmrecov. Use the same release of Backup, and install it in the same location as it was before the software was lost.

After you start the mmrecov program, the program prompts for the device from which the bootstrap save set will be extracted. Then, it asks for the bootstrap ssid. This number is found in the fourth column (labeled ssid) of the last line of the bootstrap report printed each time you run the savegrp program. Refer to the mmrecov(1m) man page for an example of the bootstrap report.

The mmrecov program works in two phases. First, it extracts the contents of the bootstrap save set, which contains the media database and online file index. The online file index contains only one entry: for itself. In the second phase, the mmrecov program runs the  recover program to completely recover the server's online file index. The final phase is performed in the background, so that you can respond to subsequent media mount requests.

The following example describes the format and options available for the mmrecov program:

```
mmrecov [-q | -v]
```

- Use the -q option to run mmrecov in quiet mode, which only displays error messages encountered.
- Use the -v option to run mmrecov in verbose mode, which displays more detail about the program's status as it is executed.

## nsrck

The nsrck program checks the consistency of the Backup online index of clients' save sets.

Use the `nsrck` to check the consistency of the Backup client file indexes. Typically, the `nsrck` program is automatically started by the `nsrindexd nsrindxd` program as part of the `nsrindexd nsrindxd` startup. The `.nsrck` file is locked upon program execution; therefore, only one instance of `nsrck` can run on the server.

You can restart the `nsrck` program at any time during its execution. Therefore, it can survive system crashes or exhaustion of resources without losing data.

Index consistency checking is done in up to four phases:

- Phase zero determines whether a client's index requires further investigation. This phase checks the internal state of the index and, if that state is consistent, avoids further passes. Phase zero also reports index names that appear to be suspicious (for example, indexes whose names do not map to valid network addresses).
- Phase one fixes any errors found in the database record file, `db`, and rebuilds the b-tree indexes for the database, if necessary.
- If you specify the `-X` option, `nsrck` invokes phase two, which cross-checks the client file index with the media database. Records that do not have existing, browsable save set entries are deleted.
- If the database requires compression, either due to space freed by the previous phases or due to a state flagged by a previous run, the index is compressed during phase three.

Index compression is a two- or three-step process. First, the records of the database are copied to a temporary database, `db.CMP`. When that operation is completed, a flag file, `db.SVC`, is created; the old, uncompressed database is removed; and the compressed database is renamed to `db`. Finally, the `db.SVC` file is removed. If there is not enough room on the filesystem containing the `db` file to include the temporary database also, `nsrck` creates a temporary file on another writable filesystem. It stores a pointer to this file in a file named `db.PTR`. In this case, an extra copy of the data is required, because the uncompressed database must first be removed before the data can be copied back to the correct place. After all these steps are completed, the `db.PTR` file is removed.

The following example describes the format and options available for the `nsrck` program:

```
nsrck [-qM] | [-T tempdir] [-X [-x percent]
| -C | -F | -m] [clientname...]
```

- Use the `-C` option to force index compression on the named clients, or all clients if none are specified. Other phases of checking are only performed if an error in a database is detected.
- Use the `-F` option to force a check on the listed client names. If no names are given, forced checks are performed for all client indexes. This option forces all phases of index checking. For backward compatibility, the `-F` option implies index compression, and may be used to force the index to be compressed. This

option is typically only necessary when the browse policy is reduced (for example, if the browse policy is changed from 1 year to 6 months). See "NSR policy" on page 289 for information on the policy resource.

- Use the -M option to use the nsrck program in master mode (not advised for manual operation). This option advises nsrck that nsrd or another Backup daemon invoked nsrck and logs messages with timestamps, as well as performs any other behavior expected by nsrd.
- Use the -m option to force nsrck to check and rebuild the media database b-tree indexes, instead of checking a client's online file index.
- Use the -q option to use nsrck in quiet mode. Quiet mode suppresses all advisory messages.
- Use the -T option to specify the directory for nsrck to use to hold the temporary database during compression, if there is not enough room in the filesystem containing the db file. If you use this option and there is insufficient space in the specified temporary directory, the nsrck program fails. This argument is ignored if there is sufficient space in the filesystem containing the db file.
- Use the -X option to tell nsrck to cross-check the ssids in the index records with save sets found in the media database instead of checking the index databases (unless an error in phase zero occurs). Records that do not correspond to media save sets are discarded. If specific clients are listed, the cross-check is limited to those client indexes.
- Use the -x option to compress a database after it has been cross-checked, if the database uses less than the specified percent of the UNIX file. The unused pages are returned back to the filesystem. The default percentage for the -x option is 30.

## nsrim

The nsrim program manages the Backup server's client file indexes and media database. Typically, the nsrim program is run automatically by the nsrmmdbd daemon when a scheduled backup starts, by the savegrp program upon completion of the backup, and by nsrd as a result of selecting the option to remove the oldest index cycle. Ordinarily, you should not run the nsrim program manually.

The nsrim program accesses the defined policies to determine how to manage each client's file index. Entries that have existed in the index longer than the period specified by the client's defined browse policy are removed from the client's file index. Save sets that have existed in the media database longer than the period specified by the client's defined retention policy are marked as recyclable in the media database. When all of the save sets contained on a single volume are marked recyclable, the volume itself is considered recyclable. Recyclable volumes may be selected and, in the case of volumes managed by an autochanger, automatically relabeled for use by Backup when a writable volume is requested for another backup. After you relabel the recycled volume, the data once contained on it is destroyed. Until you relabel the volume, you can still use the `scanner` program to recover the save sets. See “`scanner`” on page 346 for information on how to use the `scanner` program.

The following example describes the format and options available for the `nsrim` program:

```
nsrim [-b browse] [-c client] [-N saveset]
[- r retention] [-x percent] [-lnqvMX]
```

- Use the `-b` *browse* option to use the policy specified by *browse* instead of the browse policy defined in the client’s resource. This option is useful when combined with the `-n` option to determine the potential effect of a modified policy on the client file indexes.
- Use the `-c` *client* option to process only the client file index for the client specified. If you do not specify this option, all the client file indexes managed by the Backup server are processed. You can repeat multiple `-c` *client* options on the same command-line.
- Use the `-l` option to remove the oldest level full save and all save sets that depend on it from the client file index. This option is only acted on if there is more than one cycle of the save set in the client file index. This option ignores the browse and retention policies assigned to the client’s resource. The save set’s header information prints out the number of browsable full cycles that are currently maintained in the client file index. This option ignores any archive or migration save sets contained in the index. Manual save set entries are treated as if they were run as an incremental level save set. The `-l` option sets the utilization threshold to 30 percent.
- Use the `-M` option to use the `nsrim` program in master mode (which is not advised for manual operation). This option advises `nsrim` that `nsrd` or another Backup daemon invoked `nsrim`, and logs messages with timestamps, as well as performs any other behavior expected by `nsrd`.
- Use the `-N` *saveset* option to process only the named saveset; all other save sets encountered are skipped. This option may be repeated multiple times on the same command line.
- Use the `-q` option to run `nsrim` in quiet mode. This option omits the generation of header, trailer, or save set messages.
- Use the `-r` *retention* option to instruct `nsrim` to use the policy specified for retention rather than the retention policy defined by the client’s resource. This option is useful when combined with the `-n` option to determine the potential effect of a modified policy on the client file indexes.
- Use the `-x` *percent* option to set the utilization threshold. If, after removing entries, a client file index’s utilization is less than the specified amount, the value of percent is passed to `nsrindexd` `nsrindxd`  when a cross-check is requested. The default value of percent is 50. If you specify either the `-X`  or `-l` options, the utilization threshold changes to 30 percent.

- Use the -v option to run nsrim in verbose mode. This option may produce an especially large amount of output. If you specify both the -q and -v option together, the options cancel out each other's effect.
- Use the -X option to check the consistency of the save set data structures with the volume data structures. The only time you would need this option is if a Backup were to crash occur. This option sets the utilization threshold to 30 percent.

Refer to the nsrim(1m) man page for further details and a list of the most common error messages encountered.

# nsrinfo

The nsrinfo program generates reports about the contents of a client's file index. The Backup client name is required; if you provide no further options, the nsrinfo program produces a report of all the names of the files and objects, one per line, found in the *backup* namespace for the specified client. The nsrinfo program can also generate reports for a specific client file index namespace, either for all the namespaces at once, or for a particular XBSA (X-Open Backup Services) application. The report can be restricted to a single time period, called the *savetime*, which is the time the entry was entered into the client file index.

If you do not specify the -L option, you must be listed on the Backup server's Administrators list to use the nsrinfo program. If you do specify the -L option, you must be the system administrator (for example, root on a UNIX system or Administrator on a Windows NT system).

The following example describes the format and options available for the nsrinfo program:

```
nsrinfo [-vV] [-s server | -L] [-n namespace]
[- N filename] [-t time] [-X application] client
```

- The required *client* specification determines the client that the nsrinfo program is reporting on.
- Use the -L option to open the client file index directly without using the Backup server. This option is useful for debugging or to query the client file index when Backup is not running.
- Use the -n *namespace* option to specify a client file index namespace to query. By default, the *backup* namespace is queried. The other values recognized by the nsrinfo program are *migrated*, *archive* (reserved for future use), *nsr*, *informix*, and *all*.

- Use the -N *filename* option to specify an exact filename to search for in the client file index. Only index entries that are an exact match to the specified filename are printed. For some client systems (for example, NetWare), the filename stored in the client file index is often not made up of printable ASCII characters, which limits the use of this option.
- Use the -t *time* option to restrict the nsrinfo query to a single, exact save time. The value of time may be expressed in any of the Backup Backup formats. Every save set created by Backup is assigned a unique save time. You can determine the save time by using the mminfo program (see "mminfo" on page 348 for information).
- Use the -v option to instruct nsrinfo to run in verbose mode. In addition to the filename, this option displays the type of the file, any internal file index identifier designated, its size (UNIX files only), and its save time. You can combine this option with the -V option.
- Use the -V option to instruct nsrinfo to run in alternate verbose mode. In addition to the file name, this option displays the offset within the save set that contains the file, its size within the save set, the application namespace, and its save time. You can combine this option with the -v option.
- Use the -s *server* option to define the name of the Backup server that nsrinfo should query. By default, the server on the local system is queried.
- Use the -X *application-type* option to restrict the query to a list of information for a specific X/Open Backup Services (XBSA) application. Valid application types are *All*, *Informix*, and *None*. The expected value for *application-type* is not case-sensitive.

Refer to the nsrinfo(1m) man page for a full description of the valid values for *namespace*, the file types encountered in the client file indexes, an example of how nsrinfo is used, and a listing of common error messages encountered.

# nsrls

The nsrls program, when invoked without any options, prints the number of files in client file index, the number of kilobytes that the client file index currently requires, and the utilization of the client file index with respect to the number of kilobytes allocated to its UNIX file.

The following example describes the format and options available for the nsrls program:

```
nsrls [client-name...]
nsrls -f file-name...
```

- Use the *client-name* option to specify a particular Backup client file index to examine. By default, the current system is considered to be the client you want to examine indexes for.
- Use the `-f` *file-name* option to instruct `nsrls` to take a list of file names rather than a list of Backup client names. For each legitimate index file named, the `nsrls` program prints an internal volume ID number and the filename, then a statistics banner, followed by the statistics associated with each internal file in the index. Each internal file has the following statistics associated with it: an internal file ID (Fid), the number of kilobytes that the file consumes (Kbytes), the number of logical records in the file (Count), and a descriptive name for the internal file (Name). Refer to the `nsrls(1m)` man page for a full description of the internal files.

APPENDIX **C**

# Troubleshooting

If you have a problem with Backup or if the product does not work the way you expect, use the information in this appendix to diagnose your problem. The information in this appendix covers:

- Information to Gather Before You Call Technical Support
- Backup and Recover
- Client/Server Communications
- Autochanger Operations
- Backup Archive and Retrieve
- Diagnostic Tools

## Information to Gather Before You Call Technical Support

If the solutions in this appendix do not solve the problem, be prepared to provide the following information when you call Sun Technical Support:

- The software version of Backup.
- The version of operating system that you are running. For Solaris, you can determine this with the `uname -a` command. For AIX, use the `oslevel` command.
- Your hardware configuration.
- Information on your devices and other SCSI IDs. For Solaris, AIX, IRIX, and DYNIX/ptx, use the `/etc/LGTOuscsi/inquire` command as root to obtain the required information. For HP-UX, use the `/etc/ioscan` command as root to obtain the required information.
- If you are using an autochanger, the type of connection (SCSI or RS-232). Also, provide the version of the autochanger driver you are using. For Solaris, you can determine this from the output of `pkginfo -x LGTOdrvr`. For AIX, you can use `lslpp -l | grep Sun`.

You should also be able to relate the following:

- How to reproduce the problem
- The exact error messages
- How many times you have seen the problem
- Whether the Backup command was successful before you made any changes and, if so, the changes you made

# Backup and Recover

This section explains how to troubleshoot various problems you might encounter with backup and recover operations.

## Checking the Backup Daemons

If you have trouble starting Backup, the daemons might not be running properly. To determine whether the required daemons are running, enter the one of the following commands at the shell prompt:

```
# ps -aux | grep nsr or ps -ef|grep nsr
```

You should receive a response similar to the following:

```
12217 ?        S  0:09 /usr/sbin/nsrexecd -s jupiter
12221 ?        S  2:23 /usr/sbin/nsrd
12230 ?        S  0:00 /usr/sbin/nsrmmdbd
12231 ?        S  0:01 /usr/sbin/nsrindexd
12232 ?        S  0:00 /usr/sbin/nsrmmd -n 1
12234 ?        S  0:00 /usr/sbin/nsrmmd -n 2
12235 ?        S  0:00 /usr/sbin/nsrmmd -n 3
12236 ?        S  0:00 /usr/sbin/nsrmmd -n 4
12410 pts/8    S  0:00 grep nsr
```

If the response indicates that the daemons are not present, start the Backup daemons with the following commands, depending on your server:

- For Solaris and DYNIX/ptx:

```
# /etc/init.d/Backup start
```

- For HP-UX:

```
# /sbin/init.d/Backup start
```

- For AIX:

```
# nsrexecd
# nsrd
```

## Backup of Clients Fails to Stop

During a backup, you attempt to stop the process by clicking Stop in the Group Control window. This should stop the process for all clients in the selected group, but sometimes a client is missed. You then see messages that indicate the server is still busy.

To resolve the problem, on the client machine, determine which clients still have a `save` process running by using one of the following commands:

```
# ps -aux | grep save or ps -ef | grep save
```

This command returns a process identification number (PID) for each process associated with `save`. Enter the following command to stop the `save` process for each PID:

```
# kill -9 pid
```

## No Notification of Client File Index Size Growth

Backup does not notify you when a client file index is getting too large. You should monitor the system regularly to check the size of client file indexes. See “Index Management” on page 55 for information on how to manage the Backup client file indexes. See “`nsrck`” on page 356 and “`nsrim`” on page 358 for information on how to use the `nsrls`, `nsrck`, and `nsrim` programs for troubleshooting and to check the integrity of the indexes.

## Media Position Errors Encountered When Auto Media Verify Is Enabled

When you enable Auto Media Verify for a pool, Backup verifies the data written to volumes from the pool during the save. This is done by reading a record of data written to the media and comparing it to the original record. Media is verified after Backup finishes writing to the volume, which might occur when a volume becomes full or when Backup no longer needs the volume for saving data.

To verify media, `nsrmmd` must reposition the volume to read previously written data. It does not always succeed in the first attempt. These warning messages appear in the message display in the Backup administration program (`nwadmin`):

```
media warning: /dev/rmt2.1 moving: fsr 15: I/O error
media emergency: could not position jupiter.007 to file 44, record
16
```

No action is required. Backup continues to attempt to find the proper position. If Backup can find the correct position, media verification succeeds and a successful completion message appears.

```
media info: verification of volume “jupiter.007” volid 30052
succeeded.
```

In this case, ignore the earlier messages because they only indicate that Backup had problems finding the desired position on the media. If the problem is serious, media verification fails and a subsequent message gives the reason for the failure.

## PACKET RECEIVE BUFFER and NO ECB Counters Increase

When your server is waiting for a tape to be mounted or is in the process of changing an autochanger volume, you see the PACKET RECEIVE BUFFER and NO ECB counters increase on a NetWare client.

To resolve this problem, use the `nsr_shutdown` command to shut down the Backup server. Then for servers that run HP-UX 9.x, edit the `/etc/rc` file. Add the following line before the line that starts `nsrd:`

```
NSR_NO_PING=ok; export NSR_NO_PING
```

For servers that run HP-UX 10.x, edit the /sbin/init.d/Backup file. Add the following line before the line that starts nsrd:

```
NSR_NO_PING=ok; export NSR_NO_PING
```

Then for Solaris, HP-UX, and AIX, restart Backup manually. See “Checking the Backup Daemons” on page 364 for commands to restart manually.

## Backup Not Found in Expected Location for Solaris Client

On Solaris, Backup executables are installed by default in /usr/sbin. If you start a group backup on a Backup server that does not have /usr/sbin in the search path for root, the backup fails on a client that has its Backup executables in /usr/sbin. This is because the savefs command is not in the search path.

The best solution is to set the Executable Path hidden attribute for a client that has this problem. To set the Executable Path, display the Clients attribute in details view and enter the path of the executables, /usr/sbin, in the Executable Path attribute.

Another solution is to modify the search path for root on the Backup server to include /usr/sbin even if it does not exist locally.

## The scanner Program Marks a Volume Read-Only

When you use the scanner program to rebuild the index of a backup volume, the scanner program marks the volume read-only.

This is a safety feature that prevents the last save set on the backup volume from being overwritten. To write to the media without marking it read-only, use the nsrmm -o command:

```
# nsrmm -o notreadonly volume-name
```

## Index Recovery to a Different Location Fails

Suppose you attempt to recover indexes to a directory other than the one where they were originally located, and receive the following error message:

```
WARNING: The on-line index for `client-name' was NOT fully recovered.
There may have been a media error. You can retry the recover, or
attempt to recover another version of the `client-name' index.
```

Do not attempt to recover the indexes to a different directory. After the indexes have been recovered to their original location, you can move them to another directory.

Because the indexes are holey files, using the UNIX `cp` command creates a file that consumes more disk space than the original file. To move the indexes, invoke the following command as root from within the `/nsr/index` directory:

```
# uasm -s -i client-index-directory-name | (cd target-directory; uasm -r)
```

## Potential Cause for Client Alias Problems

If you encounter any of the following situations, a client alias problem might be the cause:

- You receive the following error messages: “No client resource for...” or “Client *xxx* cannot back up client *yyy* files.”
- A client machine always performs full backups, regardless of the level of the scheduled backup.
- It appears that automatic index management according to the browse and retention policies does not occur. This is indicated by the filesystem containing the indexes continuously increasing in size.
- In `/nsr/index` (the directory that contains the indexes) there are two directories for the same client using two different client names.

A client alias change is needed for the following situations:

- Machines that have two or more network interfaces
- Sites that mix short and “fully qualified” hostnames for the same machines; for example, `jupiter` and `jupiter.oak.com`
- Sites using both YP (NIS) and DNS

Use the Backup administration program or `nsradmin` to edit the client resource for clients with this problem. Add all network names for this host to the Aliases attribute.

**Caution –** Do not put aliases that are shared by other hosts on this line.

## Illegal Characters to Avoid in Configurations

When you upgrade from earlier versions of Backup, the configuration names of label templates, directives, groups, policies, and schedules that include the following special characters are no longer allowed:

```
/\\*?[]()$!^;'\"`~><&|{}
```

This change was made because volume labels, directives, groups, policies, and schedules are often passed as command line options to various Backup programs.

During installation of Backup, these characters in your current configuration names are replaced with an underscore (_) in the resources where they were originally created.

However, in the Clients resource where these configurations are applied, Backup automatically replaces the selected configuration with the preexisting Default configuration.

You need to reselect the configurations whose names have changed and reapply them to the individual clients.

## The `scanner` Program Requests an Entry for Record Size

If you use the `scanner` program with the `-s` option but without an `-i` or `-m` option, and you receive the message:

```
please enter record size for this volume ('q' to quit) [xx]
```

the number in the bracket [`xx`] is the entry from the last query.

The `scanner` command always rewinds the tape and reads the volume label to determine the block size. If the volume label is corrupted or unreadable, you see a message prompting you to enter the block size (in kilobytes).

Type in the block size; it must be an integer equal to or greater than 32. If you enter an integer that is less than 32, you receive the following message:

```
illegal record size (must be an integer >=32)
```

## Failed Recover Operation Directly After New Installation

If you attempt to start the `nwrecover` program immediately after installing Backup for the first time on your system, you receive the error message “nwrecover: Program not found.”

To save disk space, Backup delays the creation of the client index until the first backup is completed. The `nwrecover` program cannot recover data until the client index has entries for browsing. To avoid the problem, perform a backup on the client.

## File Index Is Missing Message

If you attempt to recover the file indexes with the `scanner -i` without first using `nsrck -c` to create a new index, you might encounter a message similar to the following example:

```
scanner: File index error, file index is missing.
Please contact your system administrator to recover or recreate
the index.
(severity 5, number 8)
scanner: write failed, Broken pipe
scanner: ssid 25312: scan complete
scanner: ssid 25312: 91 KB, 13 file(s)
scanner: done with file disk default.001
```

If you run the nsrck program with a -F option to check the indexes, you see messages similar to the following example:

```
nsrck: checking index for jupiter.mycompany.com
nsrck: index for jupiter.mycompany.com is missing.
nsrck: checking index for mars
nsrck: File sr does not contain any valid records.
nsrck: compressing index for mars
nsrck: WARNING no valid savetimes - cross-check not performed for
mars
```

The scanner program must have a file index to rebuild from before it can proceed. To resolve the problem, always run the nsrck program with a -c option to create an index before you run scanner -i to rebuild them from the backup media.

## How to Activate Full Diagnostic Core Dumps on AIX

Earlier releases of Backup on an AIX system do not automatically provide full diagnostic core dumps.

You need to set the NSR_FULLDUMP environment variable to 1 to activate full diagnostic core dumps. Perform the following steps to set the variable and the core file size,

1. **Become root and set the environment variable with one of the following commands as appropriate for your shell tool:**

```
export NSR_FULLDUMP=1
setenv NSR_FULLDUMP 1
```

2. **Use SMIT or edit the** /etc/security/limits **file to set the core file size to** -1 **for root or default.**

3. **Set the “Enable full CORE dump” option to True (select Change or Show Characteristics of Operating System from the System Environments choices).**

You might need to reboot your system after you edit the limits file for the core file size change to take effect. You can check the file size limit with the ulimits -a command. For the size that you set in the procedure described, the ulimits command should return a value of “unlimited.”

## “Hardware Address Not Found” Message Encountered for HPUX

The HPUX system configuration logger might generate the error message:

```
bootpd pid# Hardware address not found hardware-address
```

A similar message is written to the Backup `/nsr/logs/messages` file. If you encounter this message, perform the following steps:

1. **Become root to edit the** `/etc/syslog.conf` **file and change every instance of** `daemon.notice` **to** `local7.notice`**.**
2. **Use the** `nwadmin` **or** `nsradmin` **program to edit the Notifications resource configured for the Log Default notification and change the value shown in the Action attribute from** `daemon.notice` **to** `local 7.notice`**.**
3. **Apply and save the changes to the Log Default notification.**
4. **From the HP-UX command line, enter the following command**:

```
# cat /etc/syslog.pid
```

5. **Use the pid# obtained from the** `/etc/syslog.pid` **file to kill the designated pid# and cause the** `syslogd` **daemon to reread its configuration file**:

```
# kill -HPU pid#
```

The `local7` facility is provided as an example. View the `syslog.h` system header file to determine other possibilities for the facility. By default, `LOG_LOCAL0` to `LOG_LOCAL7` are reserved for local use. If they are not used by other local utilities on your system, Backup can make use of them exclusively to avoid the hardware address problems encountered with `bootpd`.

## Recovering Files From an Interrupted Backup

If you terminate a backup by killing the Backup daemons, you cannot recover the files because the media database is not updated when the daemons die. Consequently, Backup does not know which volumes the requested files reside on.

## Backup of a New Client Defaults to a Level Full

The first time you back up a new client, you receive the following message:

```
mars:/usr no cycles round in media db; doing full save.
```

In this example, the `/usr` filesystem on the `mars` client has no full saves listed in the media database. Therefore, regardless of the backup level selected for the client's schedule, Backup performs a full backup. This feature is important because it enables you to perform disaster recoveries for the client.

You might also receive this message if the server and client clocks are not synchronized. To avoid this, make sure that the Backup server and the client are in the same time zone and have their clocks synchronized.

## Renamed Clients Cannot Recover Old Backups

Backup maintains a client file index for every client it backs up. If you change the name of the client, the index for that client is not associated with the client's new name and you cannot recover files backed up under the old client name.

To recover previous backup data under the new client name, follow these steps:

1. **Delete the Client resource configured for the old client name.**
2. **Create a new Client resource for the new client name.**
3. **Shut down the Backup daemons using the following command:**

```
# nsr_shutdown
```

4. **Delete the index directory that was automatically created for the new client. (If you simply copy the new client index over the old client index directory, the result is a nesting of the new client index inside the old client index directory.)**
5. **Use the `mv` command to rename the old client's file index directory.**

```
# mv /nsr/index/old-client /nsr/index/new-client
```

# Disk Label Errors

If you receive the error message "No disk label," you might have incorrectly configured a nonoptical device as an optical device within Backup. Verify that the Media Type attribute in the Devices resource matches the expected media for your device, and make corrections if needed.

# Errors From Unsupported Media in HP Tape Drives

Certain Hewlett-Packard tape drives can only read 4 mm tapes of a specific length. Some, for example, read only 60 meters tapes and do not support the use of 90- or 120- meter tapes. To determine the type of tape supported by your HP drive, consult the hardware manual provided with the drive.

If you attempt to use unsupported media in an HP tape drive, you might encounter the following types of error messages in the respective situations:

- When you use the `nsrmm` or `nsrjb` command to label the tape:

```
nsrmm: error, label write, No more processes (5)
```

- When you attempt to use the `scanner -i` command:

```
scanner: error, tape label read, No more processes (11)
scanning for valid records …
read: 0 bytes
read: 0 bytes
read: 0 bytes
```

# IRIX Displays "No Space Left in Tape" Message

If you use an IRIX Backup server or storage node, you might receive the following message:

```
BSF invalid argument no space left in tape
```

This is not a Backup error but an indication that the tape drive you are using might not be configured within the machine's kernel. You need to ensure that the device is supported and configured for your machine by applying a kernel patch, available from the Silicon Graphics web site.

To resolve the problem, follow these steps:

1. **If you have not already enrolled as a SurfZone (free program) member, point your web browser to the SGI Technical Assistance Center at** `http://www.sgi.com/support/patch_intro.html` **to enroll and obtain a password.**
2. **Point your browser to** `http://support.sgi.com/surfzone/patches` **(you will need to provide your SurfZone password).**
3. **Download the recommended patch.**
4. **Follow the instructions provided to apply the patch set.**

SGI tests specific firmware revisions for each drive and robot with IRIX Backup. Even if your site uses drives and robotics that SGI has qualified, you might need to apply patches to IRIX to make the combination work.

Visit the SGI Technical Assistance Center web site for information about and distribution of patches required for your particular configuration. You can also obtain information regarding the current list of supported drives, robotics, and firmware revisions supported by IRIX.

## Cannot Print Bootstrap Information

If your server bootstraps are not printed, you might need to enter your printer's name as a hidden attribute in the Groups resource. Access the hidden attributes by selecting Details from the View menu in the graphical administration program (`nwadmin`) or by selecting the Hidden choice from the Options menu in the cursor-based administration program (`nsradmin`).

Enter the name of the printer where you want the bootstraps to be printed in the Printer attribute of the Groups resource.

## Server Index Saved

If your Backup server belongs to a group that is not enabled or does not belong to any group, Backup automatically saves the server's bootstrap information with each group that is backed up. If this is the case, you receive the following message in the savegroup completion report:

```
jupiter: index Saving server index because server is not in an
active group
```

This is a safety measure to help avoid a long recovery process in the event of a system disaster. You should, as soon as possible, configure the Client resource for the server to include it in an active backup group.

## Copy Violation

If you installed Backup on more than one server and used the same Backup enabler code for them all, you receive messages similar to the following in your savegroup completion mail:

```
--- Unsuccessful Save Sets ---
* mars:/var save: error, copy violation - servers 'jupiter' and
'pluto' have the same software enabler code, 'a1b2c3d4f5g6h7j8'
(13)
* mars:/var save: cannot start a save for /var with NSR server
'jupiter'
* mars:index save: cannot start a save for /usr/nsr/index/mars with
NSR server 'jupiter'
* mars:index save: cannot start a save for bootstrap with NSR
server 'jupiter'
* mars:index save: bootstrap save of server's index and volume
databases failed
```

To successfully rerun the backup, you must issue the `nsr_shutdown` command on each server, remove the Backup software from the extra servers, and then restart the Backup daemons on the server where you want the backups to go.

## Xview Errors

If you receive the following error message when you attempt to start the graphical administration interface with `nwadmin` from a client machine, it means that the client is not authorized to display Backup:

```
Xlib: connection to “mars:0.0” refused by server
Xlib: Client is not authorized to connect to Server
Xview error: Cannot open display on window server: mars:0.0 (Server
package)
```

To correct the situation, perform the following steps:

1. **From the client machine, invoke the** `xhost` **command:**

```
# xhost server-name
```

2. **Remotely log in to the Backup server and issue the** `setenv` **command at the shell prompt.**

```
# setenv DISPLAY client-name:0.0
```

For command shells other than `csh` enter:

```
DISPLAY=client-name:0.0
export DISPLAY
```

# Client / Server Communications

Many of the problems that Backup users report when they set up and configure Backup are problems with the communications in their networks. This section contains a procedure for testing the communications in a network.

The information provided here is for UNIX platforms only. If your Backup setup includes other platforms, refer to *Technical Bulletin* 299, which is included in the `bulletins.pdf` file that shipped with your Backup binaries.

## ▼ How to Troubleshoot IP Errors

If one of the following error conditions occur, you probably have an IP naming problem:

- RPC errors
- Unknown host messages
- Failure with contacting the portmapper
- Connection failures or timeouts
- Program unexpectedly exits
- Refused connections
- RCMD to active client fails
- Name to address translation fails
- Program not registered messages
- Backup services or daemons fail to start
- Backup services or daemons fail to remain active
- Messages about an invalid path

To troubleshoot IP errors, follow these steps:

1. **Document the steps you take and the results, especially error messages, in case you need to contact Sun Technical Support. This enables you to email or fax the exact steps and error message text directly to Sun.**

2. **Set up host tables for Backup clients and Backup servers. See "How to Set Up Host Tables" on page 378.**

3. **Disable other name servers to simplify testing. See "How to Disable Name Servers for Troubleshooting" on page 379.**

4. **Use** `ping` **to establish basic connectivity. See "How to Use** `ping` **to Verify Network Connections" on page 380.**

5. **Use** `rpcinfo` **to verify that sessions can be established and that portmapping is correct. See "How to Use** `rpcinfo` **to Verify That Sessions Can Be Established" on page 380.**

## ▼ How to Set Up Host Tables

We recommend that you troubleshoot IP problems using only host tables. Troubleshooting using only host tables does not mean you cannot use your name service, for example, DNS, with Backup. Run tests using only host tables to determine whether you have Backup installed correctly. After you know Backup works with host tables, you can enable whatever name server you are using.

To configure host tables on a server or client, follow these steps.

1. **On the Backup client, list the client and the Backup servers to which it connects, for example:**

```
127.0.0.1 localhost loopback
123.456.789.111 client client.domain.com
123.456.789.222 server server.domain.com
```

2. **On the Backup server, list the Backup server itself and all of its clients, for example:**

```
127.0.0.1 localhost loopback
123.456.789.111 server server.domain.com
123.456.789.222 client client.domain.com
```

3. **Use the guidelines in “How to Use `ping` to Verify Network Connections” on page 380 to ensure the highest success rate for host table parsing within any operating system.**

Following are some recommendationsmfor host table configuration:

- Do not use blank lines in the body of your host tables.
- The end of the host table should always contain a blank line.
- The first unremarked entry should always be the loopback line in the exact order and format shown in Steps 1 and 2.
- The last character of each unremarked line should be a space, not a carriage return.

On UNIX platforms, the host tables reside in `/etc/hosts`.

You can use host tables in addition to DNS where necessary, but it is simplest to temporarily disable DNS for troubleshooting.

## ▼ How to Disable Name Servers for Troubleshooting

To simplify the troubleshooting of name resolution problems, try disabling services like DNS, WINS, and DHCP. If you have name resolution problems, first configure only the host tables for your machines, then test your backups.

Some common problems you can encounter with DNS, WINS, and DHCP services include the following:

- The DNS is not configured with a reverse lookup table.
- The clients are configured with the wrong IP addresses for DNS or WINS servers.
- The DHCP services do not properly update the WINS server with new addresses.

You do not need to disable DNS for your entire network, just for the initial setup of the Backup clients and the Backup server you want to test. Only disable the ability of a client to obtain IP naming information from a DNS server. Typically, you do not need to disable the DNS server itself.

To disable the DNS server on most UNIX platforms, rename the file `/etc/resolv.conf` and reboot.

For a Solaris or HP-UX system, instead of renaming `resolv.conf`, you can set up the IP name search order so that the host table is searched before DNS.

To set up the IP name search order, follow these steps:

1. **Edit the** `/etc/nsswitch.conf` **file and verify that the** `/etc/resolv.conf` **file exists.**

2. **Set the host file to be first in search order, with DNS second and NIS last, for example:**

```
hosts: files [NOTFOUND=continue] DNS [NOTFOUND=continue] nis
```

For an AIX system, edit the `/etc/netsvc.conf` file and reboot.

You can also set the `NSORDER` environment variable. Refer to Info Explorer for directions specific to your version of AIX.

## ▼ How to Use `ping` to Verify Network Connections

After you create the host tables, test them with `ping`. Use just the steps marked with an asterisk (*) if the server is the only client.

On the Backup client run the following tests with the `ping` command:

- `ping` the client short name (hostname) from the client.
- `ping` the client long name (hostname plus domain information) from the client.
- `ping` the client IP address from the client.
- `ping` the server short name from the client.
- `ping` the server long name from the client.
- `ping` the server IP address from the client.

The following example shows how to ping the client short name and client long name from a Backup client called `mars` in the `oak` domain:

```
# ping mars
# ping mars.oak.com
```

On the Backup server run the following tests with the `ping` command:

- **ping** the server short name from the server.
- **ping** the server long name from the server.
- **ping** the server IP address from the server.
- **ping** the client short name from the server.
- **ping** the client long name from the server.
- **ping** the client IP address from the server.

## How to Use `rpcinfo` to Verify That Sessions Can Be Established

If `ping` is successful and backup problems still exist, you can also test with `rpcinfo`. Because Backup relies heavily on mapping of ports, use `rpcinfo` to test the operation of the portmapper. Using `ping` tests the connection up to the network layer in the OSI model; `rpcinfo` checks for communication up to the session layer.

Use the same tests with `rpcinfo` as with `ping`. Run just the steps marked with an asterisk (*) if the server is the only client.

For `rpcinfo` to be used successfully, the machine whose hostname you enter on the command line must have a portmapper running. In most cases, Sun portmappers are compatible with fully functional portmappers from other vendors (this is called a third-party portmapper). If you are using a product that provides its own portmapper, we recommend not loading the third-party portmapper until you have verified that Backup works with the rest of your environment. This process lets you test portmapper compatibility without adding other unknowns.

On Solaris, the `rpcbind` daemon must be running. On AIX and HP-UX, the `portmap` daemon must be running. The `rpcinfo` utility is part of the operating system.

The syntax for using `rpcinfo` to display ports using TCP is:

```
# rpcinfo -p hostname
```

Substitute the long name and short name for the variable *hostname*, just like for `ping`.

You can view other rpcinfo command line options by typing rpcinfo at the command line. Notes on the rpcinfo command and its error messages are available in the UNIX man page for rpcinfo. Repeat rpcinfo using all the locations and all the iterations listed in this document for ping.

When rpcinfo runs successfully, the output is a list of port numbers and names as shown in the following example:

```
rpcinfo for mars
program vers proto   port
100000    2   tcp    111  portmapper
100000    2   udp    111  portmapper
390103    2   tcp    760
390109    2   tcp    760
390110    1   tcp    760
390103    2   udp    764
390109    2   udp    764
390110    1   udp    764
390113    1   tcp   7937
390105    5   tcp    821
390107    4   tcp    819
390107    5   tcp    819
390104  105   tcp    822
```

## How to Verify Firmware for Switches and Routers

If you are using switches or routers from any vendor, make sure that the switch or router firmware is dated after August 1995 (wherever they exist on your network) to ensure that RPC (Remote Procedure Call) traffic is handled properly. Most of the switch and router vendors have significantly improved their handling of RPC traffic since August 1995.

## Naming Requirements

Backup UNIX clients, release 4.2 and later, use the servers file in the /nsr/res subdirectory to determine whether a Backup server is authorized to back up the client's data. If you don't have the servers file, you can create it in /nsr/res using your preferred editor.

Make sure the servers file on a client contains both the short name and long name of the server you want to use to back up that client's data. For example, the servers file on a Backup client would contain the following names for a Backup server named mars in the oak.com domain:

```
mars
mars.oak.com
```

In the Clients resource, list both the short name and the long name, plus any other applicable aliases for each client, in the Alias attribute.

## Binding to Server Errors

Backup follows the client/server model, where servers provide services to the client through the RPC. These services reside inside of long-lived processes, known as daemons.

For clients to find these daemons, you must register the daemons with a registration service. When the daemons start up, they register themselves with the registration service provided by the portmapper.

Backup servers provide a backup and recovery service. They receive data from clients, store the data on backup media, and retrieve it on demand. If the Backup daemons are not running and a Backup service is requested, you receive the following messages in your savegroup completion mail:

```
Server not available
RPC error, remote program is not registered
```

These messages indicate that the Backup daemons nsrd, nsrexecd, nsrindexd, nsrmmd, and nsrmmdbd might not be running. To restart the daemons, become root and enter the following command at the shell prompt:

- For Solaris:

```
# /etc/init.d/networker start
```

- For HP-UX:

```
# /sbin/init.d/networker start
```

- For HP-UX AIX:

```
# nsrexecd
# nsrd
```

## Saving Remote Filesystems

You might receive the following error messages in your savegroup completion mail when a backup for a remote client fails:

```
All: host hostname cannot request command execution
All: sh: permission denied
```

The first message means that the `nsrexecd` daemon on the client is not configured to allow the server to back up its files. The second message means that the `nsrexecd` daemon is not currently running on the client.

To resolve these problems, make sure that the `nsrexecd` daemon is running on the client, and that the server's hostname is listed in the boot-time file. The boot-time file is automatically generated before the installation script is completed, and takes your responses to the query for the names of all the servers, in order of precedence, that can contact a client for backups. Table C-1 lists the location for the boot-time file.

Refer to the `nsrexecd(1m)` man page for detailed information about the `nsrexecd` daemon.

**TABLE C-1** Boot-time File Locations

| Operating System | Boot-time file |
|---|---|
| AIX | `/etc/rc.nsr` |
| HPUX | `/etc/rc` |
| IRIX (SGI) | `/etc/rc2.d/S95networker` |
| SCO | `/etc/rc2.d/S95networker` |
| Solaris | `/etc/rc2.d/S95networker` |
| SunOS 4.1.x | /etc/rc.local |
| Ultrix | /etc/rc./local |
| others | /etc/rc2.d/S95networker |

## Remote Recover Access Rights

You can control client recover access by configuring the Client resource. The Remote Access list displays the usernames that have recover access to the client’s save sets. You can add or remove usernames depending on the level of security the files require.

The following users have permission to recover any files on any client, regardless of the contents of the Remote Access list:

- Root
- Operator
- Member of the operator group

Other users can only recover files for which they have read permission, relative to the file mode and ownership at the time that the file was backed up. Files recovered by a user other than root, operator, or the operator group are owned by that user.

# Autochanger Operations

This section explains how to resolve problems encountered with the use of an autochanger with Backup.

## Maintenance Commands

The device driver software provided with Backup provides the following maintenance commands for diagnosing problems on tape devices and autochangers:

`lusbinfo` – prints out SCSI information

`lusdebug` – sets the library debugging level

`lrescan` – rescans for devices

`lreset` – resets the SCSI bus

`changers` – lists the SCSI autochangers attached to the system

`hpflip` – flips the device type of HP optical disk drives

`ielem` – initializes the element status

`inquire` – lists the devices available

`ldunld` – loads or unloads a tape device `msense` – retrieves mode sense data

`pmode` – prints mode sense data

`relem` – reads the element status

`tur` – tests whether the unit is ready

`writebuf` – writes a device buffer

`sjiielm` – tests the standard jukebox interface (SJI) `SJIIELEM` command

`sjiinq` – tests the `SJI SJIINQ` command

`sjirdp` – tests the `SJI SJIRDP` command

`sjirdtag` – tests the `SJI SJIRTAG` command

`sjirelem` – tests the `SJI SJIRELEM` command

`sjirjc` – tests the jukebox

For more detailed information regarding these commands, please refer to their corresponding man pages.

## How to Test the Device Driver Installation

After you install the Backup device driver software, use the `lusdebug` program to verify the server connection and the `jbexercise` program to test the autochanger. Use the value of the control port assigned to your autochanger (for example, `scsidev@0.6.0`) for *control-port* in the following commands:

```
# lusdebug control-port 0
# jbexercise -c control-port -m model
```

If these commands fail or if you receive error messages, see the following sections for information on the possible cause and solution.

### The `lusdebug` Command Fails

If the `lusdebug` command fails, review these suggestions to identify the potential problems and their solutions:

- Issue the `sjiinq` command as root, and provide the `control-port` as an argument. You should receive a message similar to the following:

```
scsidev@0.6.0:<EXABYTE EXB-10i EXB-10i >
```

Verify that the information supplied by the message is correct.

If the vendor and model names are incorrect, you supplied the wrong SCSI ID as the device ID during the driver installation. The installation script asks for the SCSI ID of the robot mechanism, not the tape drive.

Uninstall the device driver and reinstall it, and supply the correct address for the autochanger (robotic arm). Make sure that each device on the SCSI bus has a different SCSI ID address.

- Inspect the following items to verify that the autochanger is properly connected:

1. Make sure all the connectors on the SCSI bus are firmly connected.

2. Make sure none of the SCSI cables are defective.

3. Verify that the SCSI bus is properly terminated and is within the length specified by ANSI SCSI-II specifications (ANSI X3.131-1994).

   Both ends of the SCSI bus must be terminated with the appropriate resistors to be properly terminated. Single-ended SCSI buses are 220 ohms to +5 VDC, 330 ohms to ground. Differential terminators have a 122-ohm characteristic impedance (-5 VDC to +5 VDC). The ends of the SCSI bus are considered to be the last SCSI device at either end of the bus, where both peripheral devices and systems are considered as peer SCSI devices.

   Additional termination (termination placed on devices not at either end of the SCSI bus) is ill-advised. Additional termination causes the hardware bus drivers on each device on the bus to work harder (for example, out of the range of their nominal specification) to affect signal transitions. As a result, they might not be able to meet the timing requirements for some signal transitions.

4. SCSI bus length limitations affect the quality of the signals, thus, the likelihood of transmission errors on the bus. For Single-ended SCSI buses (the most prevalent), the length is 6 meters, unless FAST SCSI devices are attached and in use, in which case the length limit is 3 meters. This length includes the length of the bus as it is within a device as well as the length of external cables. A reasonable rule of thumb for internal length is to assume 1 meter of internal bus length for the workstation chassis and about 0.25 meters per device for external peripheral boxes.

   Differential option SCSI buses can be much longer (due to the electrical differences from single-ended). Allow for a maximum of 25 meters. Never mix differential and single-ended devices.

- Check to see whether an old autochanger driver is still installed. This can be the AAP driver shipped with earlier versions of Backup, or release 1.1 or earlier of the Parity driver, which only supported SCSI bus 0.

Uninstall the driver according to the instructions shipped with the old driver, then reinstall the latest version. Special instructions on how to remove the AAP driver are available in Technical Bulletin 142, which is in the `bulletins.pdf` file included with the Backup software.

- Check the SCSI IDs on all devices attached to the same bus; make sure that none are the same. If two devices have the same target ID, you can see the following symptoms: SCSI bus reset errors appear in system log files, the machine does not boot, and the `probe-scsi` boot prompt command on SPARC systems hangs.
- If the sensor that verifies whether the tape drive door is open is out of place, follow the instructions provided with your autochanger hardware to determine the problem, or contact your hardware vendor.
- If the autochanger is in sequential mode, change the setting to random mode.

If none of these suggestions resolve the problem, contact Sun Technical Support. You need to provide the information described in “Information to Gather Before You Call Technical Support” on page 363 and the captured output of the `jbexercise`, `sjiinq`, and `sjirjc` programs. See Appendix B, “Command Line Reference Utilities” page 261 for information on the `jbexercise`, `sjiinq`, and `sjirjc` programs, or refer to the associated man pages for each program.

## The `jbexercise` Command Fails

If the `jbexercise` command fails, review the following list of suggestions to identify potential problems and their solutions:

- The `jbexercise` program prompts you for a nonrewinding device name (for example, on Solaris, `/dev/rmt/0mbn`). Verify that you have supplied the correct device pathname for the tape drive. The device name must belong to a tape drive in the autochanger, not the autochanger itself.

If you receive the following error message, you did not enter a nonrewinding device name:

```
device not ready
```

- Make sure that the tape drive for which you enter the pathname works. Insert a volume into the drive and perform the following tests:

  a. Use the `tar` command to copy a small file to the volume.

  b. Verify more extensive operations by issuing the `tapeexercise` command.

If these tests fail, the tape drive is not functioning. Contact your hardware vendor for further information on how to configure your tape drive to work with your system.

If none of these suggestions resolve the problem, contact Sun Technical Support. You need to provide the information described in “Information to Gather Before You Call Technical Support” on page 363 and the captured output of the `jbexercise`, `sjiinq`, and `sjirjc` programs. See Appendix B, “Command Line Reference Utilities” page 261 for information on the `jbexercise`, `sjiinq`, and `sjirjc` programs, or refer to the associated man pages for each program.

## Autodetected SCSI Jukebox Option Causes Server to Hang

If you install an Autodetected SCSI jukebox using `jb_config` and the server hangs, the following workaround is recommended:

1. **Select the `jb_config` option that installs an SJI jukebox. A list of jukeboxes is displayed.**
2. **Enter the number that corresponds to the type of jukebox you are installing.**
3. **Proceed with jb_config until you receive the following message:**

```
Jukebox has been added successfully.
```

## Autochanger Inventory Problems

The autochanger inventory becomes outdated, which means that Backup cannot use the autochanger, if any of the following situations occur:

- The media is manually ejected from the autochanger drive.
- The media is removed from the autochanger.
- The autochanger door is opened.

To make the autochanger usable again, perform the following steps:

1. **Verify that the media cartridge is correctly installed in the autochanger and that the autochanger door is closed.**
2. **Become root on the Backup server.**

3. **Reset the autochanger.**

```
# nsrjb -Hv
```

4. **Perform an inventory.**

```
# nsrjb -Iv
```

After the inventory operation is finished, Backup can once again use the autochanger.

For complete information on the use of the `nsrjb` command, refer to the `nsrjb(8)` man page or see Chapter 7, "Autochangers," on page page 153.

## Destination Component Full Messages

The message "Destination component full" usually is the result of a manual operation performed on the autochanger, for example, physically unloading the tape drive by means of the buttons on the autochanger rather than using Backup to unmount the volume. This operation causes Backup to lose track of the status of the media in the autochanger.

To resolve the problem, use Backup command `nsrjb -H` to reset the autochanger.

## Tapes Are Not Filled to Capacity

You might encounter situations where Backup does not fill tapes to capacity. For example, a tape with an advertised capacity of 4000 Mbytes can be marked full by Backup after only 3000 Mbytes of data have been written to it.

To enable Backup to use the tape capacity to its fullest, select the highest density device driver appropriate for your device. When a tape is labeled, Backup writes to it at the highest density supported by your device.

There are several reasons for situations in which Backup appears to fill tapes prematurely:

- Write errors occur during a backup.

Most tape drives try to read after a write operation to verify that the tape was written correctly, and retry if it was not. A write error indicates either and end-of-tape or read error. At any tape error, Backup marks the tape full.

To prevent tape write errors, clean your tape drive regularly and use only data-quality tapes. If cleaning the drive does not seem to help, make sure that the device driver is properly configured, any necessary switch settings on the tape drive are set to the manufacturer's specifications, all cabling is secure, and other potential SCSI problems have been addressed.

- Backup filemarks take up space on the tape.

Backup periodically writes filemarks to facilitate rapid recovery of data. These filemarks consume varying amounts of tape depending on the type of tape drive–on some drives, filemarks can consume several Mbytes. The number of filemarks Backup writes to tape is a function of how many save sets are on the tape. Many small save sets require more filemarks than a few larger ones.

- Tape capacities vary from tape to tape.

Tape capacities are not constant from tape to tape. Two apparently identical tapes from the same vendor can vary significantly in capacity. This can cause problems if you copy one very full tape to another, especially if the destination tape holds less data than the source tape.

- Data compression affects the tape capacity.

If you use compression on your tape drive, you cannot predict the effect on tape capacity. A compressing drive can provide twice the capacity of a noncompressing drive. It could be far less or far more, depending on the kind of data being backed up. For example, if a noncompressing drive writes 2 GB of data to a specific tape, the compressing drive could write 10 GB, 2 GB, 5 GB, or some other unpredictable amount of data.

- Length of tape.

Be sure to verify tape lengths. A 120-meter DAT tape holds more data than a 90-meter DAT tape, and without examining the printed information on the tape cassette carefully, the two tapes can appear identical.

Refer to Technical Bulletin 176, available in the `bulletins.pdf` file included with your Backup software distribution, for more detailed information.

For Solaris, if your tape devices are not directly supported by Sun Microsystems, you will need to recreate your entries in the *st.conf* file. If you need assistance with this, contact Sun Technical Support.

## Server Cannot Access Autochanger Control Port

The control port controls the autochanger loading mechanism. Your autochanger's hardware installation manual should have instructions on how to verify whether the control port is properly connected. If you cannot determine whether the control port is working, contact the autochanger vendor for assistance.

# Backup Archive and Retrieve

This section explains how to troubleshoot various problems you might encounter with Backup archive and retrieve.

## Remote Archive Request From Server Fails

If you cannot perform a remote archive request of a workstation from the Backup server, the archive client's user name (for example, `root`) might not be listed in that client's Archive Users attribute in the Clients resource.

You can also grant Backup administrator privileges for `root@`*`client-system`* in the Administrator attribute in the Server resource. Granting administrator privileges creates a potential security issue, since Backup administrators can recover and retrieve data owned by other users on other clients.

## Multiple Save Sets Appear as a Single Archive Save Set

When you combine multiple save sets in an archive, such as `/home` and `/usr`, they end up in a single archive save set, which appears as a slash (`/`) in the Archives list in the Backup Retrieve program (`nwretrieve`).

If you want save sets to appear separately during retrieve, archive them separately.

## Cloned Archives Do Not Appear in Backup Retrieve Window

When you search for an annotation in the Backup Retrieve program (`nwretrieve`), the Archives attribute does not display archive clones.

To locate the clones, start the query without specifying a Search Annotation attribute. If that query returns too many archives, you can use `mminfo` to locate the archive clone with the same save set ID (ssid) as the archive you want.

## Wrong Archive Pool Is Selected

If you create multiple archive pools, the one selected for archive is not the default archive pool. When you create multiple archive pools, the last one created is the one selected for archive.

## Second Archive Request Does Not Execute

If you create two archive requests with the same name, only the first request is executed. To avoid the problem, do not create two archive requests with the same name; the newer one will never be executed.

## Command Line Archive Does Not Start Immediately

If you run `nsrarchive` from the command line, the archive does not start immediately after you type the annotation and then `[Ctrl]+[D]` to start the archive. Wait a short time; there is a delay before the archive starts. Do not press `[Ctrl]+[D]` multiple times.

## Empty Annotations in Retrieve List

You might encounter empty annotations in the retrieve list when you search for annotations using a search string.

The UNIX Backup Archive program does not allow you to enter a null annotation string. By contrast, older versions of the Backup Archive software installed on DOS, Windows, and NetWare lack an annotation feature. As a consequence, the annotations for save sets archived with the older software are empty strings in the retrieve list.

# Diagnostic Tools

A variety of diagnostic tools are available as operating system services and as part of the Backup product. This section describes some diagnostic tools that are useful with Backup.

## Diagnostic Report

Backup includes a script called nsr_support that generates an exhaustive diagnostic report. Typically, you run nsr_support only at the request of Sun Technical Support. Redirect the output of the script to a file and then email the file for analysis. To run the script and redirect the output, become root on the system and enter the nsr_support command at the shell prompt:

```
# nsr_support > /temp/filename
```

For further information about the nsr_support command, refer to the nsr_support  man page.

## Communications Tests

To verify that communications sessions can be established, test with ping and rpcinfo, which are tools provided with the operating system software.

Because Backup relies heavily on mapping of ports, use rpcinfo to test the operation of the portmapper. Using ping tests the connection up to the network layer in the OSI model; rpcinfo checks for communication up to the session layer. For instructions on using ping and rpcinfo, see “Client/Server Communications” on page 377.

Contact Sun Technical Support for more tools on testing communications.

# Glossary

This glossary contains terms and definitions found in this guide. Most of the terms are specific to Backup products.

**active group** A Backup backup group that has its autostart attribute enabled.

**Administrators group** A Windows NT user group whose members have all the rights and capabilities of users in other groups, plus the capability to create and manage all the users and groups in the domain. Comparable to the superuser in a UNIX network. In Backup for Windows NT, these users are Backup administrators by default.

**annotation** A text string that the Backup administrator or user associates with a UNIX archive save set to help identify that data later. Annotations are stored in the media database for ease of searching during a retrieval operation and are limited to 1024 characters.

**Application Specific Module (ASM)** A program that, when used in a directive, specifies the way that a set of files or directories is to be backed up and recovered. For example, `compressasm` is

a Backup directive used to compress and decompress files.

**archive** The process by which Backup backs up directories or files to an archive volume and then grooms them to free disk space. When data is archived, it is written to one or more storage volumes and then marked so that it is never subject to automatic recycling. You can delete the archived files from the client, thus freeing disk space. *See also grooming.*

**archive pool** A volume pool that contains only archived save sets. A separate volume pool, the archive clone pool, contains only cloned archive save sets. Archived save sets are in a different format than regular backup save sets and must be maintained on separate media.

**archive volume** A tape or other storage medium used to store Backup archive data, as opposed to a *backup volume* or *migration store.*

**attribute** A feature of a resource. It is a service or information that the resource provides.

**autochanger** A mechanism that uses a robotic arm to move media among various components located in a device, including slots, media drives, media access ports, and transports. Autochangers automate media loading and mounting functions during backup and recovery. The term autochanger refers to a variety of robotic libraries, including jukebox, carousel, library, near-line storage, datawheel, and autoloader.

**auto media management** A feature that enables the storage device controlled by the Backup server to automatically label, mount, and overwrite a volume it considers unlabeled. Volumes that are eligible for reuse are also automatically recycled.

**authorization code** A code that is unique to your network that you obtain by sending in the registration information you print after the enabler code is entered on the Backup server. The authorization code unlocks the software for permanent use.

**backup, manual** A backup that a user requests from the client's `save` program. The user specifies participating files, filesystems, and directories. A manual backup does not generate a *bootstrap* save set.

**backup cycle** The period of time from one level full backup to the next level full backup.

**backup group** A group of Backup clients that begin their scheduled backups at the same time.

**backup levels** A measurement that determines how much data Backup saves during a scheduled or manual backup.

A full backup backs up all files, regardless of whether they have changed.

Levels one through nine (1-9) backups back up files that have changed since the last lower numbered backup level.

An incremental (incr) backup backs up only files that have changed since the last backup.

**Backup Operators group** A group of Windows NT users who can log on to a domain from a workstation or a server, back it up, and restore the data. Backup Operators can also shut down servers or workstations.

**backup volume** A tape or other storage medium used to store Backup backup data, as opposed to an archive volume or migration store.

**base enabler code** See *enabler codes*.

**bootstrap** A save set that is essential for the Backup disaster recovery procedures. It is composed of three components that reside on the Backup server: the media database, the resource database, and a server index. The server index is a file that lists all the server files that were backed up during this scheduled backup.

**browse policy** A policy that determines how long entries for your backup data remain in the client file index.

**client file index** A database of information maintained by the Backup server that tracks every file or filesystem backed up. The Backup server maintains a single client index file for each client computer.

**client-initiated backup** See *backup, manual*.

**clone** The Backup process used to make an exact copy of saved data (save sets). You can clone individual save sets or the entire contents of a backup volume. Cloning is different from a simple copy operation carried out on an operating system or hardware device because it leaves traceable information entries in both the client file index and the media data.

**clone volume** A duplicated volume. Three types of clone volumes are tracked: backup clone, migration clone, and archive clone. Save sets of different types (for example, archive and migration) cannot be intermixed on the same clone volume.

**command line interface** An interface with the Backup software, based on command text entered from the shell prompt. *See also* Appendix B, “Command Line Reference Utilities *and shell prompt*.

**DMAPI** An acronym for Data Management Application Programming Interface. The term refers to the interface defined by the XDSM Specification. *See also XDSM*.

**daemon** A program that is not invoked explicitly, but lies dormant waiting for a specified condition to occur.

**database** A collection of related data that can serve multiple purposes and support multiple users.

**DCP** An acronym for “drive control program,” which is a program that the Backup SmartMedia software uses to control the drives within an autochanger.

**device** The unit connected to the *Backup server* or *storage node*–either as a stand-alone computer or in an *autochanger*–that stores data on media.

**directed recover** A method of recovery that recovers data that originated on one client computer and re-creates it on another client computer.

**directive** An instruction that directs the Backup software to take special actions on a given set of files for a specified client during a backup.

**enabler codes** A special code provided by Sun that, when entered in the Registration resource for the Backup server, activates the software. The user must then register the software and enter the returned authorization code to permanently license the product. The enabler code that unlocks the base features for the version of the Backup software you purchased is referred to as a *base enabler*. Enabler codes for additional features or products (for example, autochanger support) are referred to as *add-on enablers*.

**filesystem** 1. A file tree that is on a specific disk partition or other mount point.

2. The entire set of all files.

3. A method of storing files.

**fingerprint** A fingerprint file is what is left on the file system after a file has been migrated by the *XDSM HSM* application. By default, the first 32KB of the file data is left with the fingerprint, the rest of the data in the file is deleted, and the associated disk blocks are freed. The reason for leaving some of the data behind in the fingerprint is for performance. Often, a user or an application will check the head of a file in order to determine if it is of interest. By leaving the first 32KB of the file with the fingerprint, the file does not have to be recalled every time the head of the file is read from or written to. Also, in some special applications, the first 32KB of the file can contain a summary of or a critical subset of the rest of the file.

**firewall** A system designed to prevent unauthorized access to or from a private network. All messages entering or leaving the intranet pass through the firewall, which examines each message and blocks those that do not meet the specified security criteria. There are several types of firewall techniques: the Backup software supports client backups from computers that are protected by *packet filtering*.

**grooming** The process of removing the original files from a local disk after a successful archive operation.

**group** A client or group of clients that starts backing up files at a designated time.

**heterogeneous network** A network with systems of different platforms and operating systems that interact across the network.

**HSM** An acronym for “hierarchical storage management,” which is a data management strategy that moves data from one storage medium to another. The hierarchy usually moves data from more expensive media with faster access to less expensive media with slower access. *See also XDSM HSM*.

**high water mark** The percentage of disk space filled that, when reached, automatically starts the migration process.

**interoperability** The capability of software and hardware on multiple computers from multiple vendors to communicate.

**LCP** An acronym for “library control program,” a program that the Backup SmartMedia software uses to control robotic libraries.

**LUS** An acronym for SunSoft User SCSI. The LUS driver is used by SunSoft software products as a proprietary device driver that sends arbitrary SCSI commands to an autochanger.

**low water mark** The percentage of disk space filled that, when reached, automatically stops the migration process.

**media** The physical storage medium to which backup data is written. Backup supports tape, magnetic or optical disk, and filesystems as backup media.

**media database** A database that contains indexed entries about the storage volume location and the life cycle status of all data and volumes managed by the Backup server.

**media manager** The Backup component that tracks save sets to backup volumes.

**migration** The process of moving data from a local filesystem to storage media in the migration store to free up disk space on the local drive.

**migration client** A filesystem on a network containing data that needs to be migrated. A migration client may consist of multiple filesystems or volumes. One or all volumes can be under migration control. The migration client must also be configured as a Backup backup client and receive migration services from a *migration server*.

**migration clone pool** A collection of storage media to which clones of migration save sets are written. *See also clone volume*.

**migration pool** A collection of storage media to which migration save sets are written. A migration pool contains data written in a different format than a backup or archive pool.

**migration server** A Backup backup server that has either the symbolic link or *DMAPI*-compliant Backup *XDSM HSM* software enabled. The migration server provides migration services to clients on a network.

**migration store** A storage device attached to the migration server that contains volumes from the migration pool.

**migration volume** The storage media that belongs to a migration pool and contains migration save sets.

**multiplexing** A Backup feature that permits data from more than one save set to be written to one storage volume on different storage devices.

**Backup client** A computer that accesses the Backup server to back up or recover data. Clients can be workstations, PCs, or fileservers.

**Backup server** The computer on a network running the Backup server software, containing the client file indexes, and providing backup and recovery services to the clients and storage nodes on the same network and media database.

**Backup storage node** See *storage node*.

**NFS client** A computer that can access files on an NFS server.

**NFS server** A computer that contains exported filesystems that NFS clients can access.

**notification** A message generated to the Backup administrator about important Backup events.

**operator** The person who monitors the server status, loads backup volumes into the server devices, and otherwise executes the day-to-day Backup tasks.

**override** A Backup feature that allows you to configure a different backup level for a specific date listed in a Schedule resource. Refer to the online help in the `nwadmin` program for instructions on how to use this feature.

**packet filtering** A method of firewall protection that looks at each packet entering or leaving the network and accepts or rejects it based on user-defined rules. *See also firewall*

**parallelism** A Backup feature that enables the Backup server to either back up save sets from several clients or many save sets from one client at the same time. Parallelism is also available during recovers.

**pathname** A set of instructions to the operating system for accessing a file. An *absolute pathname* tells how to find a file beginning at the root directory and working down the directory tree. A *relative pathname* tells how to find the file starting where you are now.

**pool** A feature that enables you to sort backup data to selected volumes. A volume pool contains a collection of backup volumes to which specific data has been backed up.

**recall** The process of copying a file from storage media in the *migration store* back to its original location on the *migration client* filesystem.

**recover** A recovery method that re-creates an image of the client filesystems and database on the Backup server.

**recyclable volume** A volume whose data has passed both its browse and retention policies and is now available for relabeling and use by a Backup server or storage node.

**Registry** A database of configuration information central to Windows NT operations. This centralizes all Windows NT settings and provides security and control over system, security, and user account settings.

**remote device** A storage device that is attached to a Backup storage node.

**resources** Anything that you might need to manage or that a user might want to locate, such as a storage device, backup schedule, or event notification. In the Backup administration program, resources are represented as windows. Resources contain attributes.

**retention policy** A policy that determines how long save set entries are retained in the Backup server's media database.

**retrieve** The process of locating and copying back files and directories that Backup has archived.

**root** 1. The UNIX superuser account (with username "root" and user ID). By extension, the privileged system-maintenance login on any operating system.

2. The top node of the system directory structure, the home directory of the root user.

**save set** A group of files or a filesystem from a single client computer backed up onto storage media.

**save set consolidation** The process that merges an incremental backup with the last full backup of a save set to create a new backup. *See also backup levels*.

**save set ID** An internal identification number that Backup assigns to a save set.

**save set recover** The recovery of specified save sets to the Backup server.

**save set status** The save set status indicates whether a given save set is restorable, recoverable, or recyclable. The save set status also indicates whether the save set has been successfully backed up.

**savestream** The data and save set information being written to a storage volume during a backup. A savestream originates from a single save set.

**server** The computer that runs the Backup software, contains the online indexes, and provides backup and recovery services to the clients on a network.

**shell prompt** A cue for input in a shell window where you enter a command. *See also command line interface*.

**silo** A repository for holding hundreds or thousands of volumes. Silo volumes are identified by barcodes, not by slot numbers. They are controlled by silo management software on a server computer that might or might not be the Backup server computer.

**SNMP** An acronym for “simple network management protocol,” which is a protocol that defines the communication between a manager (sometimes called a Monitor or Management Station) and an object (the item being managed). Backup uses SNMP to send messages to the administrator about Backup events. SNMP is not an end-user management system of its own; rather, it is the mechanism or definition that enables network management.

**staging** The process of moving data from one storage medium to another, for example, to move backed-up, archived, or migrated save sets from a disk file to a tape.

**stand-alone device** A *storage device* that contains a single drive for backing up data.

**storage device** The hardware that reads and writes data during backup, recovery, or other Backup operations.

**storage node** A storage device physically attached to another computer whose backup operations are administered from the controlling Backup server.

**versions** The date-stamped collection of available backups for any single file.

**volume** A physical unit of media, such as magnetic tape, optical disk, or disk file.

**XDSM** An acronym for X/Open Data Storage Management. For further information, refer to the X/Open web site at `http://www.xopen.com`.

**XDSM HSM** A software product that automatically moves data between a local filesystem and other storage media. The XDSM designation indicates that this version of the HSM software conforms to *XDSM* standards. *See also migration*.

# Index