NetWorker
AA-RDHLA-TE

Power Edition
Performance Tuning Guide

Digital Equipment Corporation

Maynard, Massachusetts

Operating System: Windows NT and UNIX.
Software: NetWorker for Windows NT and NetWorker for UNIX, Power Edition
Date: July 1998

Order Number: AA-RDHLA-TE

Contents

Chapter 3: Testing and Tuning on Windows NT 25

Preface

The *DIGITAL NetWorker Power Edition Performance Tuning Guide* tells you how to configure and maximize the performance of NetWorker Power Edition software.

This guide discusses the enhancements in the Power Edition product and shows you how to configure and measure the performance of your NetWorker system. Use this guide in conjunction with your NetWorker documentation, which is included in Adobe PDF format on the NetWorker product CD.

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 might also find this manual useful.

About This Guide

To use the information presented in this guide, you must install the NetWorker software on your server and NetWorker client software on your client machines. If the software is not installed, refer to the *NetWorker Installation Guide* for installation instructions.

Conventions

This guide uses the following typographic conventions and symbols to make information easier to access and understand.

- **boldface** – Indicates DOS or UNIX line commands. For example:

  Run the **jbconfig** command to add and configure the autochanger.

- *italic* – Indicates directory pathnames, files, machine names, new terms defined in the Glossary or chapter, and words or ideas that require emphasis. For example:

  Rename the original *\nsr\res* directory to *\nsr\res.orig*.

- `fixed-width` – Represents examples and information displayed on the screen. For example:

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

- Pull-down_menu>Command>Command – Depicts a path or an order to follow for making selections in the GUI. For example:

  Volume>Change Mode>Appendable

- **`fixed-width, boldface`** – Represents commands and text you type exactly as shown. For example:

  % **`dkinfo sd0a`**

- ***`fixed-width, boldface italic`*** – Represents commands and text that is replaced with user-defined text. For example:

  C:\win32app\nsr\bin **`scanner -B \\.\`*`Tape0`***

- 

  Important: Important information and cautionary notes that prevent you from making a mistake.

Chapter 1: Power Edition Features

NetWorker Power Edition is an enhanced NetWorker server product for Windows NT and UNIX operating systems. It is optimized for high-speed backup and recover operations on large amounts of complex data.

Power Edition Advantages

Power Edition addresses the storage management and data protection needs of enterprises that have high-performance database servers and file servers. To protect very large database (VLDB) applications, online transaction processing (OLTP) applications, data warehouses, and web servers, the backup software has to work quickly, often while the data and application are online and available to other users.

Power Edition takes advantage of the high throughput of the leading enterprise-class tape devices. *Performance* scales linearly as additional fast storage devices are added to the system, up to the limit of the server's I/O bandwidth.

In order to allow the server machine to maintain high data transfer rates, Power Edition uses a minimum of the processor capacity. In *benchmarking* tests, Power Edition rarely uses more than 15 percent of the CPU capacity during backup.

Performance Enhancements in Power Edition

The major performance enhancements of Power Edition include:

- Immediate save and recover

  When you back up data from the machine that is the Power Edition server or storage node, this automatic feature speeds up data transfer. CPU load is reduced because data is transferred within memory. Power Edition enables local backup and recover to bypass the CPU overhead of networking protocols.

- Storage nodes

  A Power Edition server can have several storage nodes attached, each with multiple devices, enabling the transmission of up to a maximum of 512 parallel backup sessions with a maximum of 256 devices. Because Power Edition storage nodes store data locally and maintain all of the metadata associated with the controlling Power Edition server, backup speed and efficiency is enhanced. You can use immediate save and immediate recover to back up and recover data hosted on a storage node.

- High speed device support

  Power Edition supports a broad range of autochangers and silo tape library models. Autochangers and silos can contain multiple storage devices (tape drives or optical drives). They also can automate the tasks of loading, unloading, and labeling storage media, so backups can proceed without your intervention. High speed devices are configured for higher throughput, so they can write data to media faster, in larger blocks.

- File device type and save set staging

  Support for the file device type enables backup to disk, and save set staging enables automated transfer of data from one medium to another according to user-defined policies.

- RPC (remote procedure call) enhancements

  Power Edition includes improvements to RPC, the communications between processes, to improve the efficiency of both local and remote backup and recover operations.

Power Edition reduces the time it takes to back up data in the local backup case, because the data and the Power Edition server are hosted on the same machine. Data does not have to travel through a protocol, such as TCP/IP, before it is directed to a storage device. Instead, data is transferred within memory on the server machine. Thus, network throughput is eliminated as a potential *bottleneck*. The same is true for recover operations.

When you back up data from remote clients, the data travels over the network through a protocol, such as TCP/IP. Enhancements to RPC and high speed devices improve the performance of both remote and local backups.

Architecture Shared with the NetWorker Base Product

The Power Edition server, like the NetWorker server, schedules backup of data and automates the process so that you do not have to change media or start the backup procedure manually. You can create schedules that back up your data at times when it is least used, for instance, at night or on weekends.

The Power Edition server also maintains records of the data backed up by the Power Edition server and its storage nodes, to make recover operations easier. In client file indexes, the server records the filename of each piece of data. In the media database, the server records the contents of each media volume. *Browse policies* and *retention policies* determine how long these records are kept. Backups you make on a Power Edition server or storage node can be read by a NetWorker server, and vice versa.

Power Edition also supports the file device type, for backup to disk, and save set staging, which enables automated transfer of data from one medium to another according to policies you define.

Because these functions are the same for Power Edition as for the NetWorker base product, refer to your platform's *Administrator's Guide* for more information about the scheduling, media management, and index management features of the NetWorker product line.

Overview of Performance

Performance is the speed and efficiency with which a task is completed. Backup and recover performance is dependent on several hardware and software variables, but in general, your backup can go no faster than the speed of your slowest component.

This guide provides tests you can use to measure and maximize the performance of your computer. When you find the slowest component in the data path between the location of the data and the storage device, you can change software settings or replace a hardware component to improve backup and recover performance. See "How to Test and Tune Your Power Edition Server" on page 27 for more details about how to test your computer's hardware and software performance.

Hardware performance is limited by the lowest data rate of the following computer components and related performance factors:

- Server speed
  - CPU
  - Memory
  - I/O bandwidth
  - SCSI bandwidth
  - Number of ports
- Device speed (combined for all storage devices used concurrently)
  - I/O transfer rate
  - Built-in compression and initialization characteristics

If you back up clients over a network, the following hardware performance factors can also affect performance:

- Client speeds (combined for all clients active at a point in time)
  - CPU
  - Memory
  - Disk speed
- Network speed
  - Network I/O bandwidth
  - Network path
  - Network load

Software performance factors include:

- Filesystem performance
- Application-specific optimization (see “Application Specific Tuning” on page 28 for more information)
- Backup application (for example, Power Edition)

Chapter 2: Power Edition Configuration

This chapter describes how NetWorker Power Edition works and provides examples of the following configurations:

- “Power Edition Local Backup and Recover”
- “Power Edition Backup and Recover of Remote Client Data”
- “Power Edition Storage Nodes”
- “Power Edition Local Backup to a Silo”

The second part of this chapter discusses the operations of Power Edition and the NetWorker base product in detail.

Power Edition Configuration

Power Edition is a specialized NetWorker server, so the configurations you implement for Power Edition are somewhat different than for the NetWorker server. In “How Immediate Save and Recover Works” on page 13 you can read about this special feature of Power Edition. The examples and illustrations in this section describe configuration considerations that are specific to Power Edition.

How Immediate Save and Recover Works

During a backup session on a Power Edition server or storage node, Power Edition can recognize whether the data resides on the same computer or a remote computer. When the data resides on the same computer, immediate saves and immediate recovers are invoked automatically. Instead of transferring data across the network, data is transferred within memory on the Power Edition server or storage node.

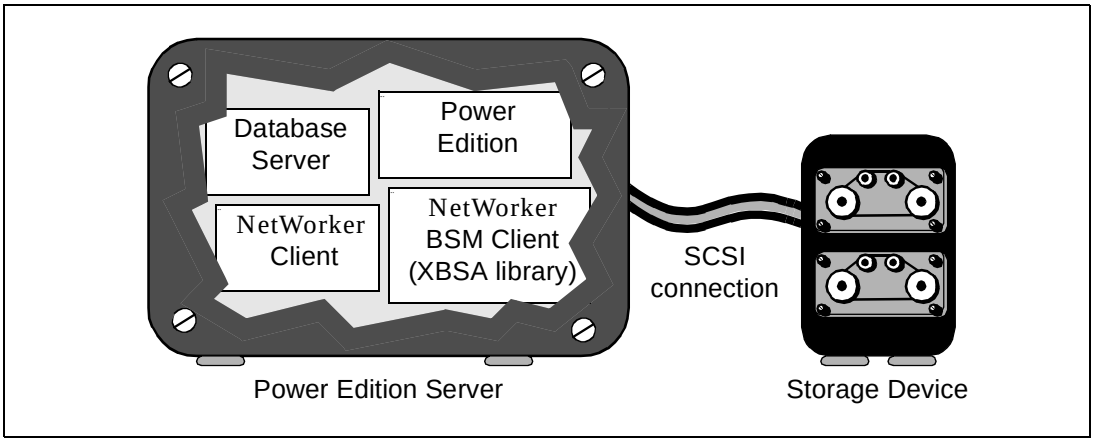
Immediate saves and recovers are available only in the case of local data transfer, where the data and the Power Edition server or storage node are hosted on the same computer. If the data is on a remote client, the data is transferred over the network using TCP/IP.

Performance for remote backup is improved through RPC enhancements. The examples in the following sections illustrate when immediate save is invoked, and when it cannot be invoked.

Power Edition Local Backup and Recover

When one computer includes both client and server or storage node software, with storage devices attached, the data is backed up using immediate save. Figure 1 shows an example of local backup of a Power Edition server. Power Edition can back up data of any type supported by a NetWorker client, including a range of databases.

Figure 1. Local Backup and Recover



Backup is faster in this setup for Power Edition than for the NetWorker base product, because Power Edition can use the immediate save technology. Processes and data that travel through a protocol stack in a standard NetWorker setup are instead passed in memory on the same computer. The network transmission bottleneck is eliminated. The same is true for recover operations.

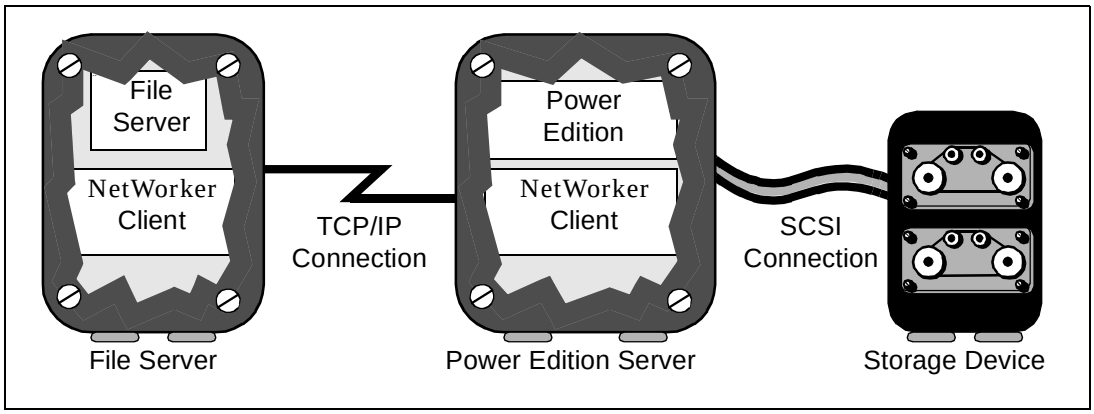
If the database is offline, any database can take advantage of immediate save for local backup with a NetWorker client. Some NetWorker BusinesSuite Module clients can also take advantage of immediate save for online database

backups. Refer to the documentation you received with your BusinesSuite Module to determine whether immediate save for online backup is currently supported.

Power Edition Backup and Recover of Remote Client Data

Figure 2 shows an example of the Power Edition server installed on a different computer than the data. To back up data from a file server of a particular platform, the NetWorker client for that platform is installed on the file server computer. The Power Edition server software and a NetWorker client (to back up data that resides on the server computer) are installed on the Power Edition server computer.

Figure 2. Remote Backup and Recover



Because the NetWorker client and the file server data reside on a remote computer, the data is transferred across the network during backup.

At recover time, the process is reversed. When the NetWorker client on the file server computer makes a recover request, the data is again transferred across the network.

Backup and recover performance over the network is enhanced for the Power Edition server compared to standard NetWorker servers, because of RPC enhancements that keep slower clients from slowing down the backup of faster clients. Data that resides on the Power Edition server computer is automatically backed up using immediate save (or automatically recovered using immediate recover).

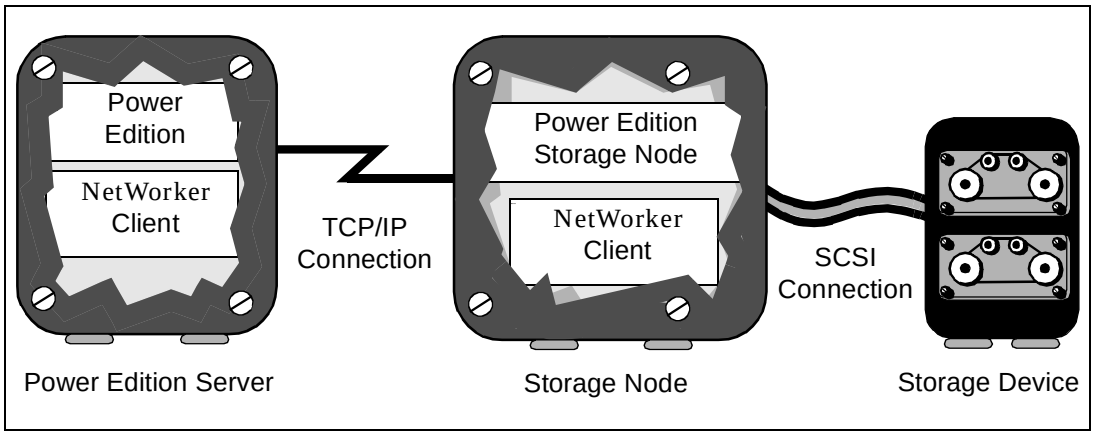
If the data is backed up to a high speed device, performance is improved because the device can transfer data faster and write data in larger blocks.

Power Edition Storage Nodes

Figure 3 shows an example of a Power Edition server using a storage node for backup and recover operations. Data hosted on each Power Edition storage node is backed up and recovered using immediate save and immediate recover technology. The metadata associated with backups is transferred over the network to the controlling Power Edition server, and the data is transferred directly to devices attached to the storage node.

Power Edition storage nodes enable you to design scalable configurations, since multiple devices can be attached to multiple storage nodes, and data transfer operations can run in parallel.

Figure 3. Power Edition Storage Node Backup and Recover

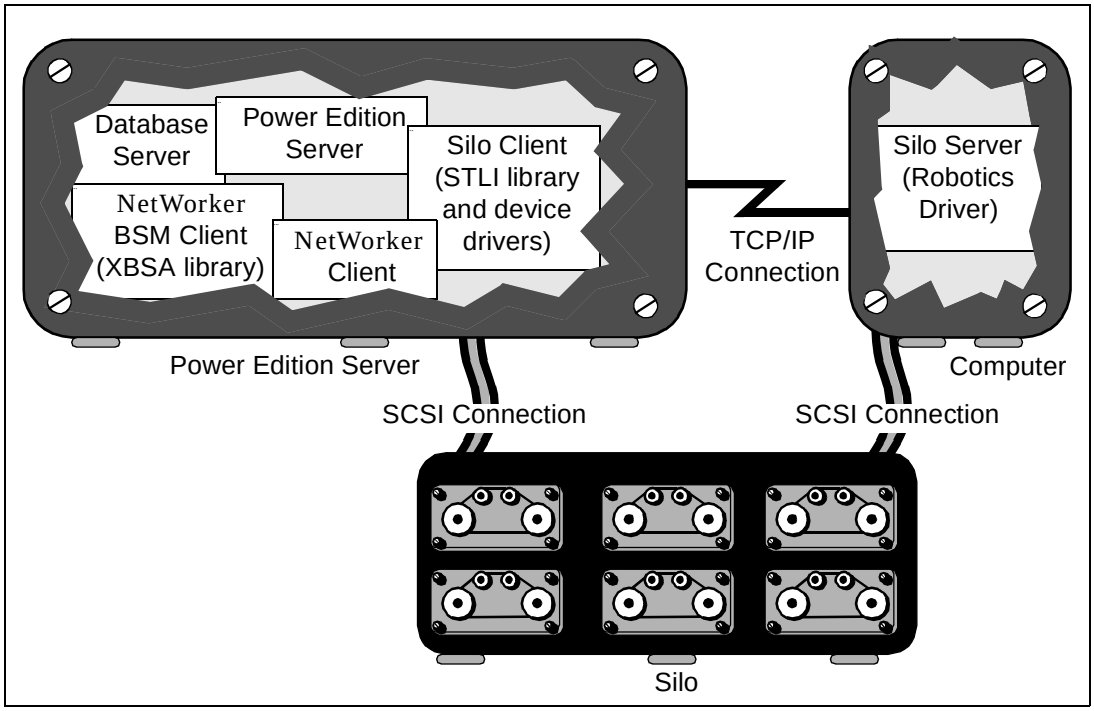


Power Edition Local Backup to a Silo

When you add a silo (specialized high-speed storage device) to the local backup scenario, performance increases because of the speed and number of devices in the silo. The silo requires another computer to be the silo server, which controls the robotics in the silo. The silo server cannot be the same computer as the NetWorker server.

In Figure 4, a silo and its silo server are added to a Power Edition local backup and recover configuration.

Figure 4. Local Backup to a Silo



During backup, immediate save is invoked to transfer data from the NetWorker client processes to the Power Edition server processes in the memory of the Power Edition server computer. If the data is written to a device in a silo, the **nsrmmd** service initiates an STLI call to the silo server, for example, to mount a tape.

The STLI call is transferred across the network connection to the silo server, then over a SCSI connection to the device in the silo. The data is transferred over a SCSI connection from the Power Edition server to the device.

During a recover, the process is similar. Media handling information is transferred across the network connection from the Power Edition server to the silo. The silo mounts the tapes, and then the data is transferred over the SCSI connection to the Power Edition server computer, where the data is transferred in memory to the NetWorker client process.

For more information about silos and how they interact with Power Edition, see the *Administrator's Guide.*

How NetWorker Power Edition Works

Because Power Edition is a NetWorker server product, it shares a similar architecture with the NetWorker base product. This section provides an overview of how the NetWorker products perform backup and recover operations.

All NetWorker servers consist of services and programs that oversee the backup and recover processes and manage storage management client configurations, client file indexes, and the media database. The NetWorker client software includes a client service and user interface programs. The services and programs in NetWorker products communicate using the RPC (remote procedure call) protocol.

NetWorker Services and Programs

The services and programs in the NetWorker software coordinate the tasks associated with a backup or recover operation, record information about what was backed up, and track the media that contains the backed-up data.

Services and Programs on the NetWorker Server

This section describes the services and programs on the NetWorker server, which contact the client for a backup and maintain the client file indexes and media database on the server. Table 1 lists these services and programs.

Table 1. Services and Programs on the Server

Service/Program	Function
ansrd	This service monitors an active **save** or **recover** session. This is an agent process spawned by **nsrd** in response to a **save** or **recover** session.
asavegrp	This service monitors the progress of individual save sets. This is an agent process invoked by the **savegrp** program.
nsrck	This service checks the consistency of the client file index. It is invoked by **nsrd** whenever the consistency of the client file index needs to be confirmed.
nsrd	This service provides an RPC-based **save** and **recover** service to NetWorker clients. This is the master NetWorker service.

Table 1. Services and Programs on the Server

Service/Program	Function
nsrim	This service automatically manages the server's media database. This service is invoked by **nsrmmdbd** when it starts up, at the end of the **savegrp** program, and by **nsrd** when a user removes the oldest backup cycle.
nsrindexd	This service provides a method for inserting entries into the client file index based on information passed by the **save** program.
nsrmmd	This service provides device support, generates mount requests, and *multiplexes* save set data during a multi-client backup. The **nsrd** service can start several **nsrmmd** services, up to twice the number of devices specified in the server.
nsrmmdbd	This service provides media database management services to the local **nsrd** and **nsrmmd** services and records entries in the media database. This is the media management database service.
savegrp	This program runs a group of NetWorker clients through the **save** process.

The NetWorker master service, **nsrd**:

- Starts other services
- Allocates media services on server computers
- Authorizes backup and recover services for the client
- Contacts clients for scheduled backups
- Maintains NetWorker configuration information
- Monitors backup and recover sessions
- Maintains server statistics and message logs

Services and Programs on the NetWorker Client

The **nsrd** service calls upon the NetWorker client service, **nsrexecd**, and several programs on the client when a scheduled or on-demand backup request is received. A temporary server agent service, **ansrd**, starts on the NetWorker server to monitor the progress of the backup session. Table 2 describes the services and programs on the NetWorker client.

Table 2. Services and Programs on the Client

service/Program	Function
nsrexecd	This service authenticates the NetWorker server's remote execution request and executes the **save** and **savefs** commands on the client.
recover	This program browses the NetWorker 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 NetWorker 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 and identifies save set data modified since the previous level **save**.

How NetWorker Backs Up Data

When you configure a *backup group* on the NetWorker server, you schedule a start time for the backup group. The **nsrd** service starts the server's **savegrp** program for the backup group at the scheduled time.

The **savegrp** program queries the client resources configured on the NetWorker 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 NetWorker server
- When the most recent backup of the group occurred

If any of this information is not available on the NetWorker server, the **savegrp** program sends a request (sometimes called a probe) to the **nsrexecd** client service 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** service to check the consistency and state of the NetWorker server's client file indexes. Then, **nsrd** starts the **nsrindexd** client file index insertion service.

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, 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 service, on the NetWorker server.

The media service, **nsrmmd**:

- Sends a message to the NetWorker server console, 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 NetWorker server's media database

Any time there is a lull in save set activity from the client, the NetWorker 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 NetWorker server's configuration, to utilize the backup devices to their maximum potential.

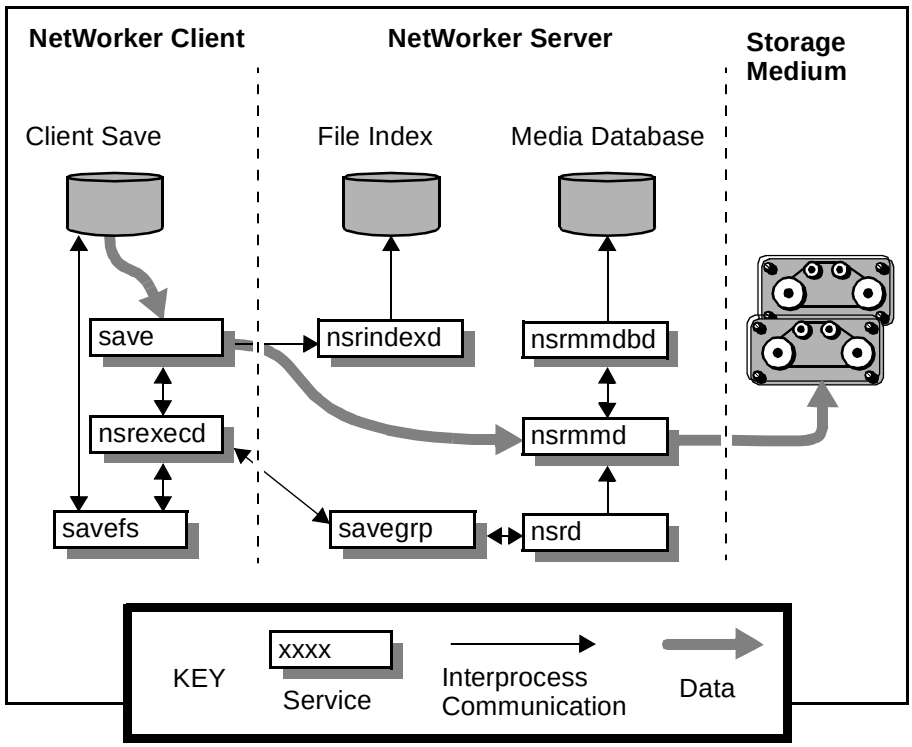
The **savegrp** program repeats the process for each item on the server's work list until all clients in the group are backed up. Before the **savegrp** program finishes, if the NetWorker server is part of the group backing up or the server is not part of any enabled group, the NetWorker server's *bootstrap* file is backed up.

If you have set up the bootstrap notification and you have installed and configured TCP/IP print services, a bootstrap printout is sent to the default printer configured for the NetWorker server after the bootstrap backup finishes. Keep the bootstrap printout in a safe place in case you need to restore the NetWorker server.

On Windows NT systems, the final results of the **savegrp** execution are returned to the server and appended to the *\nsr\logs\savegrp.log* file. On UNIX systems, the results are appended to the */nsr/logs/savegrp.log* file.

Figure 5 shows how all of the NetWorker client and services and programs interact during a scheduled save.

Figure 5. How NetWorker Services and Programs Interact during a Scheduled Backup

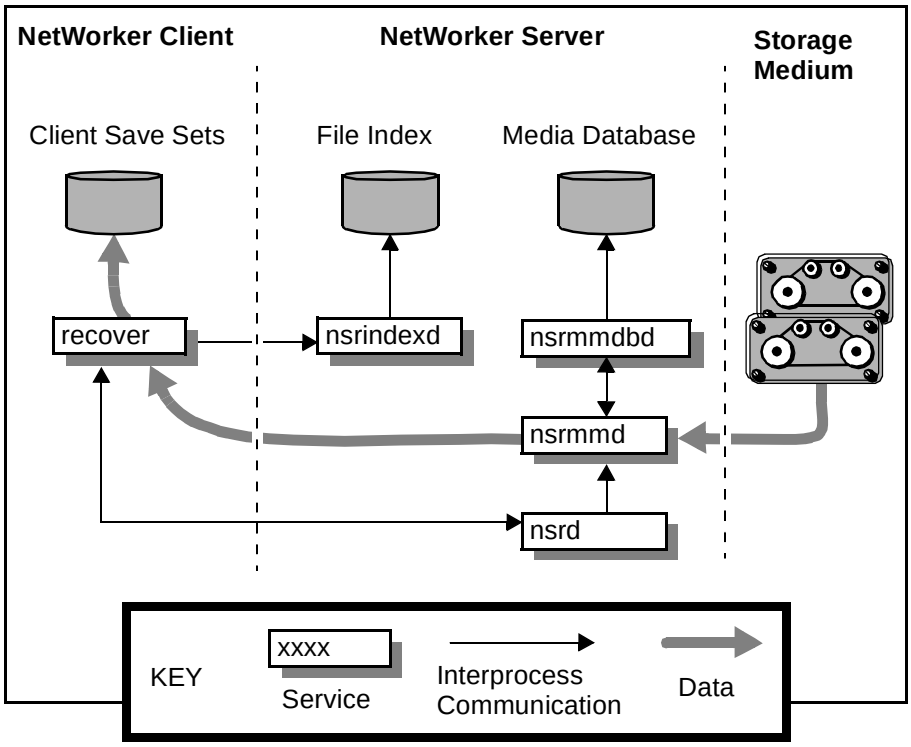


How NetWorker Recovers Data

When NetWorker receives a **recover** request from a client, the server's **nsrd** master service contacts the server's **nsrmmd** media service. The **nsrmmd** service contacts the server's **nsrmmdbd** media database service to determine which media contain the save set requested by the **recover** program. After **nsrmmd** finds the save set on media, **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 service forwards the save set to the client's **recover** program which restores the data to the client's filesystem.

Figure 6 shows how the NetWorker server and client services and programs interact while recovering data to a NetWorker client.

Figure 6. How NetWorker Services and Programs Interact during a Recover Session



Chapter 3: Testing and Tuning on Windows NT

Computer companies often publish the results of benchmarks (controlled performance tests of hardware and software) to show the potential of their products. This chapter provides the information you need to set up tests that measure the performance of Power Edition in your own environment. It also provides concrete tuning suggestions to help you get the most out of your Power Edition system.

Overview of Performance Testing

The benchmark tests used in the computer industry are done in controlled environments, to simplify analysis and show products at their greatest advantage. When you test the performance of Power Edition in your environment, you must decide how controlled a test to run.

To run a controlled test, turn off all extraneous processes to isolate the performance of Power Edition. The results of this type of test show you the optimal performance of Power Edition with your current hardware and software configuration.

To run a less controlled test, do not shut down other applications and network traffic. The results of this type of test show you how Power Edition performs while the machine is running other programs and processes.

If you run both a controlled test and a test with normal or simulated load, the difference between the results can help you decide how to schedule your backups and whether to run other programs on your Power Edition server.

The tests outlined here follow a basic strategy:

1. Measure the current performance of Power Edition system with the Windows NT Performance Monitor.
2. View the results captured by the Performance Monitor to find a bottleneck, or limiting factor.
3. Adjust the hardware and software to correct the bottleneck.
4. Measure the performance of the Power Edition system again.

Although there will always be some bottleneck in a system, you can use this process to eliminate the greatest obstacles to fast backup performance.

How to Use the Windows NT Performance Monitor

In Windows NT, use the Performance Monitor program to measure system performance.

To start the Performance Monitor in Windows NT 4.0, select Start>Programs>Administrative Tools>Performance Monitor.

In the Performance Monitor, you can choose which objects (for example, the processor or memory) to watch, which counters to measure (for example, available bytes of memory), and how to display the data. The default view of the data is a real-time line chart. However, you can also choose to log the data, format it as a report, or create alerts that warn you when a counter reaches a certain limit.

To use the Performance Monitor:

1. Select the view of the data from the View menu. The choices are Chart, Alert, Log, and Report.
2. Select the Add To Chart option from the Edit menu. The Add to Chart dialog box is displayed.
3. In the Add to Chart dialog box, specify objects for the Performance Monitor to measure.

   The sections below provide concrete suggestions of which objects and which counters to specify to measure relevant performance variables.

To measure the maximum performance of Power Edition, eliminate all extraneous processes, and run just the necessary software on your Power Edition server machine. Because the Performance Monitor uses system resources, you can run the Performance Monitor on a different Windows NT machine on the network during controlled tests.

When you run the Performance Monitor on a different Windows NT system, the Performance Monitor still uses some resources on the Power Edition system. It especially adds to network traffic during remote backup. But the performance of the processor and SCSI throughput on the Power Edition system are not affected.

For example, if your Power Edition server machine is called *jupiter*, you can measure its performance from another Windows NT machine, called *mars*, on the same network.

1. Start the Performance Monitor on *mars*.
2. Select the data view, for example, Report, from the View menu.
3. Select Add To Report from the Edit menu. The Add to Report dialog box is displayed.
4. In the Add to Report dialog box, enter **jupiter** in the Computer field.
5. Select the objects and counters you want to measure.

For more detailed information about the Performance Monitor and Windows NT, refer to the documentation provided by Microsoft.

How to Test and Tune Your Power Edition Server

When you test the performance of your Power Edition system, you measure the performance of all the variables that can affect performance, including:

- Hardware components of the Power Edition server machine
- Device hardware
- Software, including the Windows NT operating system, Power Edition, and the database software

This section describes the many factors that contribute to performance for Power Edition for Windows NT, how to measure these performance variables, and how to tune the performance of each component.

Disk

The physical disks in a computer system read data to be backed up and write data to memory, or to the network. Measure the performance of the disks where your data resides, on your Power Edition server and remote clients, to determine whether a bottleneck exists that keeps you from fast backup performance.

Data

The type of data you back up and how it is laid out on disk can cause backup performance to vary. If you have large files that are fragmented, or if you have many small files, the disk performance suffers, because the disk head has to move to go from one fragment or file to another. Backup is most efficient for large files with minimal fragmentation.

If the data is compressed on the disk, Windows NT decompresses the data before transferring it for backup. The CPU spends cycles to decompress the files, and the disk can rarely go at its maximum speed.

Application Specific Tuning

You can use database structure layout and specific backup configuration parameters to optimize application-specific backup performance. For specific information, refer to the documentation provided by the application vendor.

How to Measure Disk Performance

In the Performance Monitor, select the Physical Disk object and the Percent Disk Time and Disk Queue Length counters. If your machine has more than one physical disk, select these counters for *each disk*. Then start a backup and capture the results.

The following situations indicate that the disk is a bottleneck:

- If the value for the Percent Disk Time counter exceeds 90 percent

  This counter measures the percentage of operating time your disk is occupied with read / write requests.

- If the value for the Disk Queue Length counter exceeds 2

  This counter measures the average number of read / write requests that were waiting to be serviced during the sampling interval.

Important: If the disk performance counters do not work, enter the following Windows NT command to turn on the performance counters: **diskperf -y** Note that you must reboot your computer after using this command.

How to Tune Disk Performance

If you find a performance bottleneck in your disk system, you can make the following hardware changes to improve performance:

- Defragment files to reduce the number of disk seeks.

- Use larger files or fewer files to reduce the number of disk seeks.
- Change to a faster SCSI adapter, with synchronous I/O support and bus mastering.
- Use stripe sets on multiple disks to accommodate large files and databases, so the physical disks can work in parallel to read data.
- Spread disk-intensive programs onto different server machines.
- Purchase disks with higher transfer rates.

Memory

In general, a Windows NT machine can never have too much memory. Data transfer within memory is fast. Memory holds frequently used data so the disk does not have to be read as often. Open applications, running processes, and disk cache all use memory. When you run out of physical memory, Windows NT uses disk space as *virtual memory.*

How to Measure Memory Performance

Measure memory use on your Power Edition server to determine whether memory is a constraint to fast backup performance. In the Performance Monitor, select the Memory object. Select the Available Bytes, Pages/sec, Committed Bytes, and Commit Limit counters. Then, start a backup and capture the results.

The following situations indicate that memory is a bottleneck:

- If the value for Available Bytes is less than 4 MB

  This counter measures how much memory is not already occupied. The smaller the value of Available Bytes, the slower your performance.

- If the value for Pages/sec is greater than 20

  This counter measures virtual memory activity; that is, how often the memory writes to disk. When your machine is memory-bound, it pages to the virtual memory on disk more often.

- If the value for Committed Bytes is greater than the amount of physical memory on your machine

  This counter is the total of all memory that all applications are actually using at the moment. Applications generally reserve more memory than they use during regular operation. But an application does not write data to memory without committing the memory first. So the value for Committed Bytes shows the amount of memory the applications currently need.

If more memory is committed than you have physical memory to accommodate, your machine pages memory contents to disk (virtual memory), and your machine works more slowly, because physical memory is faster than virtual memory.

- If the value for Committed Bytes consistently exceeds the value for Commit Limit

  The Commit Limit counter is the size of virtual memory that can be committed without having to extend the paging file. Every time Windows NT extends the paging file, it goes to disk and searches for more space to use for virtual memory. This process has a high performance cost, and you might run out of disk space.

How to Tune Memory Performance

To adjust the allocated size of virtual memory:

1. In the Control Panel, double-click the System icon. The System Properties dialog box is displayed.
2. In the System Properties dialog box, click the Performance tab.
3. On the Performance page, click the Change button. The Virtual Memory dialog box is displayed.
4. In the Virtual Memory dialog box, change the size and location of the disk space allocated to virtual memory.

   To determine a good baseline size for the virtual memory file (*pagefile.sys*), log the Committed Bytes counter over a period of time, add 10 to 20 percent to the maximum value, and enter that value in the Initial Size (MB) attribute in the Virtual Memory dialog box.

To adjust memory use on a Windows NT server:

1. In the Control Panel, double-click the Network icon, then click the Services tab.
2. On the Services page, select Server, then click the Properties button. A dialog box is displayed, with the following options:

Table 3. Memory Optimization Options

Optimization Option	Purpose
Minimize Memory Used	Use when the server has fewer than 10 users, for example, Power Edition local backup.

Table 3. Memory Optimization Options

Optimization Option	Purpose
Balance	Use when the server has 10 to 64 users.
Maximize Throughput for File Sharing	Allocate most of the memory to the file server module, for systems with more than 64 users.
Maximize Throughput for Network Application	Use for a client-server application server machine, such as a NetWorker server with remote clients.

The Windows NT file server module requires a lot of memory. If your Power Edition server machine is not also a file server in Windows NT, you can select a different option to reduce operating system memory use.

To speed up virtual memory, you can implement the disk recommendations in "How to Tune Disk Performance" on page 28. You can also defragment the disks where *pagefile.sys* resides. It is not recommended to put *pagefile.sys* on a stripe set, because virtual memory is less efficient when it is fragmented. It is also not recommended to put *pagefile.sys* on a mirrored drive, because every write is done twice, which slows down writes to virtual memory.

You can also reduce the memory your Power Edition server uses. Suggestions to reduce memory requirements include:

- Reduce the number of other applications that run on the Power Edition server. Dedicate your Power Edition server to run the database server, NetWorker client, BusinesSuite Module client, and Power Edition server.
- Stop services that you do not use. For example, if you have only SCSI devices, stop the ATDISK service, which is for IDE devices only.
- Remove communications protocols you do not use. For example, if you only use TCP/IP but you also have SPX installed, remove SPX.

CPU

Each backup session requires CPU and memory resources on your Power Edition server. Other applications running concurrently with the backup impose additional load on the system. Heavy swapping or paging activities are indicative that the server is CPU- or memory-bound.

How to Measure CPU Performance

To determine whether another CPU or a CPU upgrade would improve your backup performance, measure total CPU utilization during backup.

1. In the Performance Monitor, select the Processor object.
2. If your Power Edition server has more than one processor, select each processor from the Instance list.
3. Select counters to measure, for example, % Processor Time.

   You know you have a CPU bottleneck if total CPU utilization remains above 90% for long during the backup process.

You can also measure CPU utilization for each process involved in the backup, to determine which processes use most of the CPU capacity. The processes involved in a Power Edition backup are:

- NetWorker Power Edition services
  - **nsrexecd**
  - **nsrd**
  - **nsrmmd**
  - **nsrmmdbd**
  - **nsrindexd**
- **save**, the Power Edition backup program
- Application-specific services, such as for a database server

In the Performance Monitor, select the Process object. One by one, select each of the processes involved in a Power Edition backup from the Instance list, and select the % Processor Time counter for each. Then start a backup and capture the results.

How to Tune CPU Performance

To increase CPU bandwidth, upgrade the CPU or add more CPU modules to your Power Edition server. You can also shut down other processes during backup time to see if that reduces the processor load and improves backup performance.

I/O

The following performance variables relate to the input/output (I/O) throughput on your Power Edition server:

- I/O backplane

  On a well-configured Power Edition server, the maximum data transfer rate of the server's I/O backplane is the limiting factor for backup throughput. The theoretical backup throughput limitation is the maximum I/O throughput divided by two:
  `Backup throughput = Maximum I/O throughput/2`

  At minimum, Power Edition requires two I/O transactions per data block: read from the disk to memory, and write from memory to media.

- SCSI I/O bandwidth

  SCSI bus performance depends on the technology deployed. Some specifications for different types of SCSI technology are:

  – SCSI-2 single-ended, 10MB/s
  – FWD (fast-wide-differential), 20MB/s
  – UltraSCSI, 40 MB/s

  Note, however, that:

  1. The rated speeds of SCSI buses are defined at best-case.
  2. The selection of UltraSCSI-compatible storage devices is currently limited.

How to Measure I/O Performance

In the Performance Monitor, select the System object, then select the File Write Bytes/sec counter, and run the following tests:

- To measure how fast your system can read from disk (disk I/O), use the **uasm** module provided with NetWorker. For example, enter the following command at the command prompt:

  `uasm -s filename > NUL`

  The **uasm** module reads from the disk at the maximum speed. In this example, the data is written to a null location so the disk read is not slowed down to allow time for writing. Replace *filename* with the name of the file for **uasm** to read.

- To measure SCSI throughput and the write speed of a tape drive or optical drive (device I/O), use the **bigasm** module provided with NetWorker. The **bigasm** module generates a file of the specified size, transfers it over a SCSI connection, and writes it to a tape or optical device.

To set up a **bigasm** test:

1. Create a directive file, *nsr.dir*, that contains the following command:

   bigasm -S*size* : *filename*

   For example:

   bigasm -S100M : bigfile
2. Create a file with the name you specified in your directive, for example, *bigfile*, in the same directory as your directive.
3. Back up the file using Power Edition. For example, if your file is called *bigfile* and your Power Edition server machine is called *jupiter*, enter:

   save -s jupiter bigfile

For more information about the **uasm** module and the **bigasm** module, see "uasm" on page 37.

To determine if you have a bottleneck, compare the disk read speed to the device write speed.

– If the disk can read faster than the device can write, the bottleneck is in the device or the SCSI transfer.
– If the device can write faster than the disk can read, the bottleneck is disk speed.

- The amount of data a SCSI device can write to each block of media is limited by the amount of data that you can transfer over the SCSI bus in a single I/O operation. To find the limitation of your SCSI adapter, use *MT.EXE*, a program that comes with NetWorker.

  Enter **mt -f *device* stat** at the command prompt, and look at the maximum block size value in the output. The maximum block size value is the maximum media block size you can use. For detailed information about *MT.EXE*, see "mt" on page 40.

  For example, you enter:

  ```
  D:\>mt -f \\.\Tape0 stat
  ```

  and the output is:

  ```
  \\.\Tape0:
  Media Capacity  = 2.12GByte
  Media Remaining = 3.06GByte
  Media Block size = 0
  Media Partition Count = 0
  ```

```
Media is not write protected
default block size = 8192
maximum block size = 1040384
```

The maximum block size value limits the maximum number of bytes per SCSI transfer on disk reads and volume writes.

How to Tune I/O Performance

The amount of data a SCSI device can write to each block of media is limited by the amount of data that transfers over the SCSI bus in a single I/O operation.

To tune SCSI performance, change the number of memory pages the SCSI host bus adapter (HBA) can scatter/gather in a single direct memory access (DMA). In Windows NT, the number of bytes that can be transferred in a single SCSI I/O is limited by the number of memory pages that the SCSI host bus adapter can scatter/gather in a single DMA. Each vendor's SCSI driver specifies the maximum number of pages that the HBA can scatter/gather for the DMA transfer. When you multiply the page size by the number of scatter/gather entries, the product is the effective limit to the number of bytes that can be transmitted in a single I/O operation. The equation is:

```
I/O operation size=Page size * # of scatter/gather entries
```

For the drivers of some SCSI host adapters, you can change the maximum number of scatter/gather pages. Many tape drives and disk drives perform better with a larger I/O limit. A change in the number of scatter/gather pages can make a large performance difference. For example:

- The Ampex DST tape drive writes only 3 MB per second with 64 KB SCSI transfers and 15 MB per second with 992 KB SCSI transfers.
- The SCSI disk "CYBERNET 10XP" reads 1.7 MB per second with a 32 KB read buffer size and 17 MB per second with a 200 KB read buffer size.

To change the maximum scatter/gather pages setting in the Windows NT Registry, add a new Registry subkey to specify the maximum number of scatter/gather list elements for each device on a given bus.

Important: Modifying the Registry is very dangerous. Do not attempt to modify the Registry unless you have an up-to-date backup of the Registry to which you can revert and you are convinced that the modification you are about to make is correct. Inappropriate changes to the Registry can result in the need to re-install Windows NT.

Add the subkey entry, as follows:

1. Open the Registry with the appropriate program, usually the following:

   %SystemRoot% \ system32 \ regedt32.exe

2. Within the directory tree on the left side of the HKEY_LOCAL_MACHINE window, select:

   SYSTEM \ CurrentControlSet \ Services

3. Look for the appropriate driver for your SCSI controller; for example, AIC78xx for Adaptec's 294x / 394x series.

4. Use the Edit pulldown menu to create the following subentry:

   \HKEY_LOCAL_MACHINE*SYSTEM*\CurrentControlSet\
   Services***DriverName***\Parameters\Device***n***\MaximumSGList

   where:

   – ***DriverName*** is the name of the miniport driver, such as AIC78xx.
   – ***n*** (in **Device*n***) is the bus number assigned at initialization.

   If a value is defined for MaximumSGList in this subkey when the device initializes, the SCSI port driver uses MaximumSGList as the initial value for NumberOfPhysicalBreaks. The miniport driver can set NumberOfPhysicalBreaks to a lower value, if appropriate. The maximum value for MaximumSGList is 255, but use this adjustment sparingly because scatter / gather lists consume nonpaged memory.

5. After you change the Registry entry and reboot the machine, enter **MT -f \\.\Tape0 stat** again to display the new value for maximum block size.

Other suggestions for tuning I / O performance include:

- Move the hardware configuration around; for example, move some disks off a SCSI bus that is overloaded.
- If possible, use PCI adapters in your Power Edition machine. If you must use other adapters, EISA adapters are better than ISA adapters, because ISA adapters have bus contention problems that degrade system performance.
- Use a dual PCI backplane system, rather than a bridged PCI backplane.
- Use multiple SCSI buses with devices of the same type (for example, UltraSCSI) on the same bus.
- Use short, high-quality cables with active terminators.
- Ensure that SCSI communications are occurring synchronously, not asynchronously.

uasm

The **uasm** module is used in NetWorker to save and recover filesystem data. The syntax is as follows:

uasm -s [**-benov**] [**-ix**] [**-t** ***time***] [**-f** ***proto***] [**-p** ***ppath***] ***path***

uasm -r [**-dnv**] [**-i** {**nNyYrR**}] [**-m** ***src*** **=** ***dst***] [**-z** ***suffix***] ***path***

uasm -c [**-nv**] ***path***

The **uasm** module is built into the **save** and **recover** programs. You can call **uasm** directly, in a manner similar to **tar**.

Important: The description and options of **uasm** applies to all other ASMs, including **bigasm**.

The **uasm** module has three modes: save, recover, and compare.

- In save mode, **uasm** walks directory trees and generates a save stream on its **stdout** representing the files and organization of the directory tree. Symbolic links are never followed by ASMs, except **rawasm**.
- In recover mode, **uasm** reads a save stream from its **stdin** and creates the corresponding directories and files.
- In compare mode, **uasm** reads a save stream from its **stdin** and compares the save stream with the files already in the filesystem.

In save mode, **uasm** can be controlled by directive files. Directive files control how descendent directories are searched, which files are ignored, how the save stream is generated, and how to process subsequent directive files.

All ASMs accept the options described below. These options are generally referred to as the standard-asm-arguments. ASMs can also have additional options. A particular ASM's additional options must be capital letters.

Either **-s** (save), **-r** (recover) or **-c** (compare) mode must be specified and must precede any other options. When saving, at least one ***path*** argument must be specified. The ***path*** argument can be either a directory or a filename.

Options Valid for all Modes of uasm

- Use the **-n** option to perform a dry run. When saving, walk the file system but do not attempt to open files and produce the save stream. When recovering or comparing, consume the input save stream and do basic sanity checks, but do not actually create any directories or files when recovering or do the work of comparing the actual file data.
- Use the **-v** option to Turn on verbose mode. The current ASM, its arguments, and the file it is processing are displayed. When a filtering ASM operating in filtering mode (that is, processing another ASM's save stream) modifies the stream, its name, arguments and the current file are displayed within square brackets.

Options for Save Mode

- Use the **-b** option to produce a byte count. This option is like the **-n** option, but byte count mode estimates the amount of data to be produced, instead of actually reading file data, so this option is faster but less accurate than the **-n** option. Byte count mode produces three numbers: the number of records, (files and directories), the number of bytes of header information, and the approximate number of bytes of file data. Byte count mode does not produce a save stream, so its output cannot be used as input to another ASM in recover mode.
- Use the **-o** option to produce an "old style" save stream that can be handled by older NetWorker servers.
- When you specify the **-e** option, **uasm** does not generate the final "end of save stream" boolean. This flag should be used only when an ASM invokes an external ASM and when an optimization chooses not to consume the generated save stream itself.
- Use the **-i** option to ignore all save directives from *.nsr* directive files found in the directory tree.
- The **-f** *proto* option specifies the location of a *.nsr* directive file to interpret before processing any files. Within the directive file specified by *proto*, <<path>> directives must resolve to files within the directory tree being processed, otherwise their subsequent directives will be ignored.
- When you specify **-p** *ppath,* this string is added to the beginning of each file's name as it is output. This argument is used internally when one ASM exec's another external ASM. The path you enter for *ppath* must be a properly formatted path which is either the current working directory or a trailing component of the current working directory.
- When you specify **-t** *date*, you set the date after which files must have been modified before they are saved.

- Specify **-x** to cross filesystem boundaries. Normally, filesystem boundaries are not crossed during walking. Symbolic links are never followed, except in the case of **rawasm**.

Options for Recover Mode

- Specify **-i {nNyYrR}** to define the initial default overwrite response. Only one letter is allowed. When the name of the file being recovered conflicts with an existing file, the user is prompted for overwrite permission. The default response, selected by pressing [Return], is displayed within square brackets.

  Unless otherwise specified with the **- i** option, **n** is the initial default overwrite response. Each time a response other than the default is selected, the new response becomes the default.

  When you specify either **N**, **R**, or **Y**, you are prompted only when NetWorker cannot auto-rename a file that already ends with the rename suffix, and each subsequent conflict is resolved as if the corresponding lower-case letter had been selected.

  The valid overwrite responses and their meanings are:

  – **n**

    Do not recover the current file.

  – **N**

    Do not recover any files with conflicting names.

  – **y**

    Overwrite the existing file with the recovered file.

  – **Y**

    Overwrite files with conflicting names.

  – **r**

    Rename the conflicting file. A dot, ".", and a suffix are appended to the recovered file's name. If a conflict still exists, NetWorker prompts you again.

  – **R**

    Automatically renames conflicting files by appending a dot, ".", and a suffix. If a conflicting filename already ends in a "." suffix, NetWorker prompts you to avoid potential auto rename looping conditions.

- Use the **-m** *src=dst* option to map the filenames to be created. Any files that start exactly with *src* are mapped to have the path of *dst* replacing the leading *src* component of the pathname. This option is useful if you want

to perform relocation of the recovered files that were saved using absolute pathnames into an alternate directory (for example, **-m c:\win32app=.**).

- Enter **-z** *suffix* to specify the suffix to append when renaming conflicting files. The default suffix is **R**.

mt

The **mt** program sends commands to a magnetic tape drive.

`mt -f tapename command... [ count ]`

Replace the *tapename* variable with the name of a raw tape device, for example, \\.*Tape0*.

By default, **mt** performs the requested operation once; to perform the operation more than once, specify a value in place of *count*.

The available commands are listed below:

- **eof, weof**
  Write count EOF marks at the current position on the tape.
- **fsf**
  Forward space over count EOF marks. The tape is positioned on the first block of the file.
- **fsr**
  Forward space count records.
- **bsf**
  Back space over count EOF marks. The tape is positioned on the beginning-of-tape side of the EOF mark.
- **bsr**
  Back space count records.
- **nbsf**
  Back space count files. The tape is positioned on the first block of the file. This is equivalent to count + 1 bsf followed by one fsf.
- **asf**
  Absolute space to count file number. This is equivalent to a rewind followed by a fsf count.

For the following commands, count is ignored:

- **eom**
  Space to the end of recorded media on the tape.This is useful for appending files onto previously written tapes.

- **rewind**
  Rewind the tape.
- **offline, rewoffl**
  Rewind the tape and, if appropriate, take the drive unit off-line by unloading the tape. It cycles through all four tapes.
- **status**
  Print status information about the tape unit.
- **retension**
  Rewind the cartridge tape completely, then wind it forward to the end of the reel and back to beginning-of-tape to smooth out tape tension.
- **erase**
  Erase the entire tape.
- **format**
  Low-level format the tape.

The **mt** program returns a 0 exit status when the operations were successful, 1 if the command was unrecognized or if **mt** was unable to open the specified tape drive, and 2 if an operation failed.

Not all devices support all options. Some options are hardware-dependent.

Devices

Devices are peripherals used to read or write data on storage media, such as tape drives and optical drives. The devices you use with Power Edition affect performance. Device performance variables include:

- I/O Transfer Rate

  This is the rate at which data can be written to the storage device. Depending on the device and media technology, device transfer rates can range from 500 KB per second to 15 MB per second. Default block size and buffer size of a device affect its transfer rate.

- Built-in compression

  Turn on a device's compression to increase effective throughput to the storage device. Some storage devices have a built-in hardware compression feature. Depending on how compressible the backup data is, this can improve effective data throughput, from a ratio of 1.5:1 to 3:1.

How to Test Device Performance

To test device I/O, see the **bigasm** test described in "How to Measure I/O Performance" on page 33.

How to Tune Device Performance

Most tape drives come with hardware data compression enabled by default. When compression is enabled, device performance and media capacity can double. For data that is already compressed before it reaches the device, however, additional compression might *increase* the size of the data. If the data comes to the device already compressed, you might want to disable compression on the storage device.

Hardware data compression is controlled by the NSR_NO_HW_COMPRESS environment variable. When you set this environment variable to yes, data is not compressed by the hardware before it is written to media.

To change environment variable settings in Windows NT:

1. In the Control Panel window, double-click the System icon.
2. In the System window, select the Environment tab.
3. The Environment page shows several System Variables. If the variable you want to change is listed, double-click it and edit its values in the Variable and Value fields. If the variable is not listed, enter the name of the variable in the Variable field and the value you want to assign the variable in the Value field.

   For example, enter **NSR_NO_HW_COMPRESS** in the Variable field and **yes** in the Value field.
4. Reboot Windows NT so your changes take effect.

To see the device compression setting, enter **MT -f \\.\Tape0 stat** at the command prompt. In the output of a device with compression enabled, you see a line that says Compress Enabled.

Windows NT Software

In the Windows NT software, the following variables affect the overall operating system performance and the speed of backup:

- Filesystem overhead

  Filesystem I/O can degrade I/O performance. If you integrate logical volume managers with the filesystem, you can increase disk performance and add a host of other features to improve manageability of large filesystems. Also, depending on the level of RAID (Redundant Array of Independent Disks) you implement, RAID can improve or adversely affect backup performance.

- Server memory usage setting

  You can set Windows NT server memory use to be optimized for a small number of clients, a large number of clients, a file server, or a domain controller. See Table 3, "Memory Optimization Options," on page 30 for more information.

- Page file size

  You can adjust the size and location of the virtual memory file (*pagefile.sys*) on your local disks. See "How to Tune Memory Performance" on page 30 for more information.

Power Edition Software

The following settings in Power Edition software adjust the speed of backup:

- Server parallelism

  The Parallelism attribute in the Set Up Server dialog box controls how many *savestreams* the server accepts at the same time.

  The more savestreams the server can accept, the faster the storage devices and clients' disks run, up to the limit of their performance or the limits of the connections between them.

- Multiplexing

  The Target Sessions attribute in the Edit Device dialog box sets the target number of savestreams to write to a device at the same time. Because this value is not a limit, a device might receive more sessions than the Target Sessions attribute specifies.

  The larger the number of sessions you specify for Target Sessions, the more save sets are multiplexed, or interleaved, onto the same volume.

- Client parallelism

  The Parallelism attribute in the Edit Client dialog box controls how many savestreams a client can send at the same time. The more savestreams a client sends to a server during backup, the faster its disks run. To avoid disk contention, do not set a value for client parallelism that is higher than the number of physical disks on the client.

- Immediate save and immediate recover

  The Power Edition server selects immediate save technology automatically for local backup and recover operations. There are no settings to adjust for immediate save or immediate recover.

- File device type and staging

  Power Edition includes the file device type and save set staging features, which you can use to improve backup performance. Use the file device type feature to direct your backups to disk media, instead of tape or optical media, since reads and writes to disk are generally faster than to tape or optical media.

  Use save set staging in conjunction with backups to the file type device. Save set staging lets you move save sets you have backed up from one medium to another, according to criteria you set, such as space remaining on the disk or the age of the save set. With staging, you can automate moving older backups from the file type device to make space for new backups.

  A possible configuration to maximize performance of file type devices would be:

  1. Create one directory per user, preferably with a mnemonic name matching a client.
  2. Create one file device type per directory, setting the volume size to 1GB.
  3. Create one client per system, setting the device that is always used to have the matching mnemonic device name.

  This configuration limits your system to 64 users with Power Edition, since 1 user = 1 device, and that is how many devices Power Edition supports. However, if you use storage nodes, then you can have multiples of 64 devices.

  If you back up to file type devices, you get high-speed backups to a hard drive in near real time, but the disk tends to fill quickly with backups. Without save set staging, you would have to closely monitor disk usage to avoid running out of space for your backups, and you would either have to move save sets manually or create very short browse and retention policies.

  With save set staging, the process of moving data from disk storage to tape or optical is automated, and the space that was occupied by the save set is freed to make space for new backups. The browse and retention policies for the save sets on tape or optical can be as long as you like. You can also use save set staging to move files from other types of media, but the space is only reclaimed on the file type device.

  If you are worried about disk contention when you back up to file type devices, you can use many smaller drives (for example, 6 drives of 4 GB each and 4 users per drive, instead of one 23GB drive).

You can also use a RAID array, which provides redundancy in case of a drive failure. Windows NT includes a software RAID capability that works well, but it uses many CPU cycles. A hardware RAID solution is more efficient, though more expensive.

How to Test Power Edition Software

To determine whether to multiplex data onto your storage media:

1. Find the backup rate of each disk on the client. Use the **uasm** test described in "How to Measure I/O Performance" on page 33.
2. Find the maximum rate of each device. Use the **bigasm** test described in "How to Measure I/O Performance" on page 33.

   If the sum of the backup rates from all disks involved in a backup is greater than the maximum rate of the device, do not increase server parallelism. If you multiplex more savegroups in this case, backup performance does not improve, and recover performance could slow down.

To verify that the immediate save function is working when you expect, run a local backup in debug mode, and look at the output for "using immediate save."

For example, on a machine called *mars*, you enter:

```
C:\nsr\bin> save -D 1 -s mars c:\autoexec.bat
```

And the output is:

```
save: using `C:\AUTOEXEC.BAT' for `c:\autoexec.bat'
save: ssid 56420 using immediate save to `mars'
asm -s C:\AUTOEXEC.BAT
save: access(C:\nsr\debug\NSR_NTBUFFERED) sets
NT_buffered to 0
asm -s C:\
save: mars.legato.com:C:\AUTOEXEC.BAT size 3 KB, 2
file(s), took   0 min 2 sec
save: C:\AUTOEXEC.BAT 3 KB 00:00:09      2 files
save completion time:  7-24-97  5:25p
```

During a successful immediate save backup, the Performance Monitor shows low CPU utilization and no local network traffic. For instructions about how to measure CPU utilization, see "How to Measure CPU Performance" on page 32.

How to Tune Power Edition Software

- Set the server Parallelism and Target Sessions attributes so that the total of the performance of the disk drives equals the total performance of the tape drives. If you set the Parallelism attribute to a higher value, there is no benefit.

  To select the right values for the Target Sessions and Parallelism attributes, use the following equation:

  ```
  Parallelism = Number of Devices * Target Sessions
  ```

  For example, if you have three tape drives available for backup, and you want each tape drive to accept two savestreams, set the value of server parallelism to 6 and the value of Target Sessions to 2.

- Decrease the server Parallelism and Target Sessions attributes to unload an overworked Power Edition server. With the right settings, the machine's normal operation is not interrupted by backups or other Power Edition activities.
- Add more memory to handle a higher parallelism setting.
- Balance the number of disks and devices, so the backups finish at about the same time. (For instructions on how to change these settings, refer to the *Administrator's Guide*.)
  - In general, start backups for the slowest or the biggest disk volumes first, because other disks can join in later to match the maximum bandwidth of the device.
  - Use pools of media to force the bigger and faster disk volumes to back up to faster devices.
  - If you have a very large and fast disk volume and many tape drives, you can manually divide the volume into several save sets, such that each of them can go to each tape drive in parallel.
  - Balance data load for simultaneous sessions more evenly across available devices by adjusting the Target Sessions attribute. This parameter specifies the minimum number of save sessions that must be established before Power Edition attempts to assign save sessions to another device.
- To fully use the bandwidth of a high-speed device when you have many slow clients or disks, multiplex the save sets on media. You maximize the performance of your devices when they do not have to start and stop to wait for data.

- To improve recover performance, multiplex save sets that you will recover together onto the same tape. For example, multiplex the disks of the same machine onto the same media, because it is likely that the data would be recovered at the same time.

In cases where you back up data across a network, the following suggestions can also improve performance:

- Increase the client Parallelism attribute, especially for clients with a logical volume manager and several physical disks. The bandwidth of your network could limit the number of savestreams you can transmit at one time.
- To reduce server load, split backups across multiple servers.
- Turn on data compression in the NetWorker client to improve effective data throughput and reduce network traffic.
- Configure clients with high transfer rate requirements to have a preference for a particular backup server on the same subnet; avoid router hops between the NetWorker server and its clients.

Network Hardware

If you back up data from remote clients, the routers, network cables, and network interface cards you use affect the performance you get. This section lists the performance variables in network hardware and suggests some basic tuning for networks. Network performance variables include:

- Network I/O bandwidth

  The maximum data transfer rate across a network rarely approaches the manufacturer's specification because of network protocol overhead. For example, a 10BaseT network is nominally rated at 10 mbits/sec (1.25 MB/sec). TCP packet overhead can reduce this figure by half. You can use FDDI and 100 Mbit Ethernet to expand network bandwidth, but TCP packet overhead might reduce this figure.

- Network path

  Networking components such as routers, bridges, and hubs consume some overhead bandwidth, which degrades network throughput performance. To improve performance, minimize the number of network components in the data path between the client and server.

- Network load

  Other network traffic limits the bandwidth available to Power Edition and degrades backup performance. As network load reaches a saturation threshold, data packet collisions degrade performance even more.

How to Tune Network Hardware

- Add additional network interface cards to client systems to expand network bandwidth available to clients.
- Upgrade to faster LAN media, such as 100 Mb Ethernet or FDDI, to improve network bandwidth.

Chapter 4: Testing and Tuning on UNIX

Given the multitude of platform- and network-specific variables that can impact UNIX system performance, it is not possible to discuss them all comprehensively in a single chapter. Instead, the scope of this chapter is limited to the detection and analysis of processes that have an impact on the ability of Power Edition to efficiently perform backups.

While there are no simple solutions to performance issues, this chapter provides a general testing strategy for locating performance bottlenecks. The testing strategy uses benchmarks and baselines derived from common UNIX utilities.

A system administrator can use the testing strategy to develop and refine a platform- and network-specific strategy for a particular system configuration. The same strategy can be applied to determine how variables not discussed in this chapter might impact a network's performance.

This chapter includes the following information:

- Testing strategy for detecting performance bottlenecks
- Scripts for collecting benchmarks and determining baselines
- Factors that impact system performance
- Tactics for improving system performance

Assumptions

The information in this chapter assumes that you are familiar with the UNIX system infrastructure including networking and devices. In addition, you should be familiar with writing UNIX shell scripts and using backup utilities such as **tar**.

The scripts in this chapter use standard UNIX utilities. The examples are specific to Solaris, but the scripts can be applied to most UNIX platforms.

The testing strategy and scripts make the following assumptions:

- All backup processes are local (that is, not over a network).
- All tape devices operate at the same speed.
- Data is stored on a file system (no databases are used).

Strategy for Detecting Performance Bottlenecks

The following steps provide a general testing strategy for detecting performance bottlenecks. The sections that follow describe how to apply this strategy for measuring the read speed of disks and the write speed of tape drives.

1. Determine and record the baseline for each individual device (or client) using the software currently installed on your computer, *without* NetWorker.
2. Determine and record the baseline for all devices (or clients) using the software currently installed on your computer, *without* NetWorker.
3. Determine and record the baseline for each individual device (or client) using the software currently installed on your computer, *while* NetWorker is running.
4. Determine and record the baseline for all devices (or clients) using the software currently installed on your computer, *while* NetWorker is running.

How to Determine Disk Performance

Apply the testing strategy in the previous section to collect baselines to measure disk performance:

1. Measure the read speed of each individual disk, without NetWorker.
2. Measure the read speed of all disks, without NetWorker.
3. Measure the read speed of each individual disk, while NetWorker is running.
4. Measure the read speed of all disks, while NetWorker is running.

Measuring the Read Speed of a Single Disk

To measure the read speed of an individual disk, follow these steps:

1. Use the **df** command to display a list of mounted filesystems and their respective sizes.
2. Run the **/bin/time** utility for each disk. For example, on Solaris, enter:

   `/bin/time tar cvf - `*`mount_point`*` > /dev/null`

   To achieve a consistent result, no other disk or system activity should be running when you run this command.

Measuring the Read speed of several disks

After you execute the **/bin/time** utility for each individual disk, measure the read time for all the disks on your SCSI channel. If you are running the disks in parallel, develop a baseline for the number of parallel disks. For example, if there are 300 disks, and 10 are in parallel, measure baselines for the 10 disks.

To streamline this task, create a shell script using the editor of your choice. Following is an example shell script from a Solaris computer that was created for multiple disks (*/space, /space2, etc.*):

```
/bin/time tar cvf - /space > /dev/null &
/bin/time tar cvf - /space2 > /dev/null &
/bin/time tar cvf - /space3 > /dev/null &
...
```

Measuring the Read Speed of a Single Disk While Running NetWorker

To measure the read speed of an individual disk using NetWorker, follow these steps:

1. Use the **df** command to display a list of mounted filesystems and their respective sizes.
2. Run the **uasm** command for each disk. For example, on Solaris, enter:

   `/bin/time uasm -s `*`mount_point`*` > /dev/null`

   Depending on the configuration of your computer, you may need to specify the path for **uasm** in the command above.

Measuring the Read Speed of Several Disks While Running NetWorker

Measure the read speed for all the disks on your SCSI channel. If you are running the disks in parallel, develop a baseline for the number of parallel disks. For example, if there are 300 disks, and 10 are in parallel, measure baselines for the 10 disks.

To streamline this task, create a shell script using the editor of your choice. Following is an example shell script for a Solaris computer that was created for multiple disks (*/space, /space2, etc.*):

```
/bin/time uasm -s /space > /dev/null &
/bin/time uasm -s /space2 > /dev/null &
/bin/time uasm -s /space3 > /dev/null &
...
```

Evaluating Your Benchmarks

The UNIX utility **/bin/time** reports the amount of time required to execute a program, breaking down the total time into three components: real time, user time, and system time. Real time is the time that the program runs as it would be measured by a user sitting at the terminal using a stopwatch. User time is the actual time that the computer spent executing code in the user state. System time is the time the computer spent executing UNIX system code on behalf of the user.

For example, the result of running the **/bin/time tar cvf - /space2 > /dev/null** command on a Solaris computer named *passport* provides the following result for the mount point */space2*:

```
RESULTS:
real     6:53.4
user       18.2
sys      1:48.1
```

To calculate the read speed of your disk, follow these steps:

1. Use the **df** command to obtain the file size (in kilobytes) of */space2*:

   ```
   % df
   Filesystem         kbytes    used    capacity   Mounted on
   /dev/dsk/c0t1d0s0  1952573   797935  46%       /space2
   ```

2. Divide the real time by the size (in kilobytes) of */space2*.

   For example, based on the result and size of */space2* above, the read speed of */space2* is 4727 kb / sec.

Compare the results of the benchmarks generated from individual disks without NetWorker to the results of individual disks running NetWorker. Then, compare the results of the benchmarks generated from multiple disks without NetWorker to the results of multiple disks running NetWorker.

How to Determine Drive Performance

Apply the testing strategy on page 50 to collect baselines to measure drive performance:

1. Measure the write speed of each individual tape drive, without NetWorker).
2. Measure the write speed of all tape drives, without NetWorker.
3. Measure the write speed of each individual tape drive, while NetWorker is running.
4. Measure the write speed of all tape drives, while NetWorker is running.

Measuring the Write Speed of a Single Drive

To measure the write speed of a single drive, run the following commands for each drive:

```
mt -f /dev/rmt/drive_name rewind
dd if=/dev/zero of=/dev/rmt/drive_name \
bs=block_size count=1
/bin/time dd if=/dev/zero of=/dev/rmt/drive_name \
bs=block_size count=1000
```

Measuring the Write Speed of Several Drives

Measure the write speed of all the tape drives on your SCSI channel. To streamline this task, create a shell script using the command above. Following is an example of a shell script for a Solaris computer that was created for multiple drives (*0mbn, 1mbn, 2mbn, etc.*):

```
mt -f /dev/rmt/0mbn rewind; dd if=/dev/zero \
of=/dev/rmt/0mbn bs=32k count=1; \
/bin/time dd if=/dev/zero of=/dev/rmt/0mbn \
bs=32k count=1000 &

mt -f /dev/rmt/1mbn rewind; dd if=/dev/zero \
of=/dev/rmt/1mbn bs=32k count=1; \
/bin/time dd if=/dev/zero of=/dev/rmt/1mbn \
bs=32k count=1000 &

mt -f /dev/rmt/2mbn rewind; dd if=/dev/zero \
of=/dev/rmt/2mbn bs=32k count=1; \
/bin/time dd if=/dev/zero of=/dev/rmt/2mbn \
bs=32k count=1000 &
...
```

Measuring the Write Speed of a Single Drive While Running NetWorker

To measure the write speed of a single drive while running NetWorker, follow these steps:

1. Set target session to 1.
2. Label the tape.
3. Set up **bigasm**:
   a. Create an empty directory:
      `mkdir /empty`
   b. Go to the directory *empty:*
      `cd /empty`
   c. Create an empty file:
      `touch e`
   d. Create an *.nsr* file with a **bigasm** directive:
      `bigasm -S100M : e`
4. Run the following script to determine the baseline:

   ```
   save -s <server_name> /etc/motd; /bin/time save -s \
   <server_name> -f /empty/.nsr /empty/e
   ```

   Note: In the command above, there is a space between */.nsr* and */empty*.

NetWorker automatically assigns each command to a tape, since target sessions are set to 1.

Measuring the Write Speed of Several Drives While Running NetWorker

To measure the write speed of several drives, repeat steps 1-4 in "Measuring the Write Speed of a Single Drive While Running NetWorker" for every drive on the SCSI channel.

Evaluating Your Benchmarks

Compare the results of the benchmarks generated from individual drives writing dating without NetWorker to the results of individual drives writing data when NetWorker is running. Then, compare the results of the benchmarks generated from multiple drives writing data without NetWorker to the results of multiple drives writing data while running NetWorker.

Factors That Impact Performance

This section outlines the hardware and software factors that can affect system performance of a server, network, and storage devices.

Server Performance

I/O Backplane

On a well-configured Power Edition server, the maximum data transfer rate of the server's I/O backplane is the limiting factor for backup throughput. The theoretical backup throughput limitation is the maximum I/O throughput divided by two:

Backup throughput = Maximum I/O throughput/2

At minimum, Power Edition requires two I/O transactions per data block:

- A read from the disk to memory
- A write from memory to media

SCSI I/O Bandwidth

SCSI bus performance depends on the technology deployed. Some specifications for different types of SCSI technology are:

- SCSI-2 single-ended, 10 MB/s
- FWD (fast-wide-differential), 20 MB/s
- UltraSCSI, 40 MB/s

Note, however, that:

- The rated speeds of SCSI buses are defined at best-case.
- The selection of UltraSCSI-compatible storage devices is currently limited.

CPU and Memory Capacity

Each backup session requires CPU and memory resources on your Power Edition server. Other applications running concurrently with the backup impose additional load on the system. Heavy swapping or paging activities indicate that the server is CPU- or memory-bound.

Network Performance

Network I/O Bandwidth

The maximum data transfer rate across a network rarely approaches the manufacturer's specification because of network protocol overhead. For example, a 10BaseT network is nominally rated at 10 mbits/sec (1.25 MB/sec).

TCP overhead can reduce this figure. You can use FDDI and 100 Mbit Ethernet to expand network bandwidth, but TCP overhead similarly reduces this figure.

Network Path

Networking components such as routers, bridges, and hubs consume some overhead bandwidth, which degrades network throughput performance. To improve performance, minimize the number of network components in the data path between the client and server.

Network Load

Other network traffic limits the bandwidth available to Power Edition and degrades backup performance. As network load reaches a saturation threshold, data packet collisions degrade performance even more.

Storage Device Performance

I/O Transfer Rate

This is the rate at which data can be written to the storage device. Depending on the device and media technology, device transfer rates can range from 500 KB per second to 15 MB per second. Default block size and buffer size of a device affect its transfer rate.

Built-in Compression

Some storage devices have a built-in hardware compression feature. Use built-in compression to increase effective throughput to the storage device. Depending on the backup data is, compressibility can improve effective data throughput from a ratio of 1.5:1 to 3:1.

Drive Streaming

To obtain peak peformance from most storage devices, stream the drive at its maximum sustained throughput. Without drive streaming, the drive must stop to wait for its buffer to refill or to reposition the media before the drive can resume writing. This can cause a delay in the cycle time of a drive, depending on the storage device.

Tactics for Improving System Peformance

Server Tuning

- Set parallelism and sessions per device so that the total performance of the disk drives can equal the total performance of the tape drives. Setting parallelism higher than this number does not yield any benefit.

- Decrease parallelism and sessions per device to unload an overworked server. By optimizing this setting, the effect on the server's normal operation can be minimized.
- Add more memory to handle a higher parallelism setting.
- Increase CPU bandwidth by upgrading or adding more processor modules.
- Reduce server load by splitting backups across multiple servers or storage nodes.

Network Tuning

- Add more network interface cards to client systems to expand network bandwidth to clients.
- Upgrade to a faster LAN medium such as 100 Mb Ethernet or FDDI to improve network bandwidth.
- Configure clients that have high transfer rate requirements to show a preference for a particular backup server (or storage node) on the same subnet; avoid router hops between NetWorker server and clients.

Device Tuning

- Add more tape drives or upgrade to higher speed tape devices for increased throughput.
- Balance data load for simultaneous sessions more evenly across available devices by adjusting sessions per device. This parameter specifies the minimum number of save sessions that must be established before NetWorker attempts to assign save sessions to another device.
- Turn on hardware compression to increase effective throughput to the storage device.

Glossary

This glossary contains terms and definitions found in this manual. Most of the terms are specific to NetWorker products.

1-9	Intermediate backup levels. Each number represents a backup level. Lower levels back up more files.
8mm	Choice in the Devices window; represents eight-millimeter magnetic tape.
active group	An active group is one with its autostart attribute enabled.
Administrators group	Members of this Windows NT user group have all the rights and abilities of users in other groups, plus the ability to create and manage all the users and groups in the domain. Only members of the Administrators group can modify Windows NT OS files, maintain the built-in groups, and grant additional rights to groups.
annotation	A comment that you associate with an archive save set, to help identify that data later on. Annotations are stored in the media index for ease of searching 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.

archive	The process by which NetWorker backs up directories or files to an archive volume and then optionally deletes them to free up disk space.
archive clone pool	A volume pool composed exclusively of archive clone save sets.
archive pool	A volume pool composed exclusively of archive save sets.
archive volume	A tape or other storage medium used for NetWorker archives, as opposed to a backup volume.
ASM	*See Application Specific Module.*
attribute	An attribute is a property of a resource, shown in the graphical user interface as an entry field, push button, or a list from which you select. *See also,* resource.
auto media management	A feature that enables NetWorker to automatically label, mount, and overwrite a volume it considers unlabeled. NetWorker also automatically recycles volumes eligible for reuse.
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 backups and recovers. The term autochanger refers to a variety of backup devices, including jukebox, carousel, library, near-line storage, datawheel, and autoloader.
backup cycle	The period of time from one level full backup to the next level full backup.
backup group	A group of NetWorker clients that begin their scheduled backups at the same time.

backup levels	A backup level determines how much data NetWorker saves during a scheduled or manual backup. The backup levels are: **full** - backs up all files, regardless of whether they have changed. **levels 1-9** - back up files that have changed since the last lower numbered backup level. **incremental (incr)** - backs up only files that have changed since the most recent backup.
backup volume	A tape or other storage medium used to store NetWorker backup data, as opposed to an archive volume or migration store.
benchmarking	Tests in a controlled environment to measure the performance of a product as it performs specific tasks.
Backup Operators group	Members of this Windows NT group have the capability to log onto a domain from a workstation or a server, back it up, and restore the data. Backup Operators also can shut down servers or workstations.
bootstrap	The bootstrap save set is essential for NetWorker disaster recovery procedures. It is composed of three components that reside on the NetWorker server: the media database, the resource database, and a server index which is a file that lists all the server files that were backed up during this scheduled backup.
bottleneck	The slowest point in a system, which restricts the system from operating any faster.
browse policy	A policy that determines how long entries for your backup data remain in the client file index.
carousel	A tray or tape cartridge that holds multiple backup volumes.

client file index	A database of information maintained by the NetWorker server that tracks every file or filesystem backed up. The NetWorker server maintains a single client index file for each client machine.
clone	The process NetWorker uses to make an exact copy of saved data (save sets). NetWorker can clone individual save sets or the entire contents of a backup volume. Cloning is different from an operating system or hardware device copy operation because it leaves traceable information entries in both the client file index and the media data.
clone volume	A duplicated volume. NetWorker can track three types of clone volumes: backup clone, migration clone, and archive clone. Save sets of different types cannot be intermixed on the same clone.
command-line interface	An interface with the NetWorker software, based on command text entered from the shell prompt. *See also shell prompt*.
compressasm	A NetWorker directive used for compressing and decompressing files.
database	A collection of related data that can serve multiple purposes and support multiple users.
device	The unit connected to the NetWorker server, either as a standalone machine or in an autochanger, that stores data on media.
directed recover	A method of recovery that recovers data and re-creates it, either on the machine where the backup originated, or on the machine where the administrator requests the directed recover.
directive	An instruction directing NetWorker to take special actions on a given set of files.

enabler code	A special code provided by DIGITAL that activates the software. To permanently license the product, you must then register the software and enter the returned authorization code.
fileserver	A machine with disks that provides services to other machines on the network.
filesystem	1. A file tree which is on a specific disk partition or other mount point (for example, the C: drive). 2. The entire set of all files. 3. A method of storing files.
file index	A database of information maintained by NetWorker which tracks every file or filesystem backed up.
full (f)	A backup level in which all files are backed up, regardless of when they last changed.
grooming	The process of removing original files from the 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.
himt	A choice in the Devices window; represents half-inch magnetic tape.
immediate save	A technology employed in NetWorker that makes the local backup process faster. Data is transferred in memory instead of through a networking protocol.
incremental (i)	A backup level in which only files that have changed since the last backup are backed up.
interoperability	The ability of software and hardware on multiple machines from multiple vendors to communicate meaningfully.

level [1-9]	A backup level that backs up files that have changed since the last backup of any lower level.
machine	Any computer, including file or computer servers, workstations, or diskless workstations.
manual backup	A type of 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.
media	The physical storage medium to which backup data is written. NetWorker supports tape, file, and magnetic or optical disk as backup media.
media database or media index	A database that contains indexed entries about the storage volume location and the lifecycle status of all data and volumes managed by the NetWorker server.
media manager	The NetWorker component that tracks save sets to backup volumes.
multiplexing	A NetWorker feature that permits data from more than one save set to be interleaved and then written to one storage volume.
NetWorker	The DIGITAL network-based software product for backing up and recovering filesystems.
NetWorker client	A machine that accesses the NetWorker server to back up or recover data. Clients can be workstations, PCs, or fileservers.
NetWorker server	On a network running the NetWorker server software, the machine containing the client file indexes and providing backup and recover services to clients and storage nodes on the same network and media database.
notice	A response to a NetWorker event.

notification	A message generated by the NetWorker server to alert the administrator when the NetWorker server needs attention, for example, to clean a device or mount a tape.
nsrhost	The logical *hostname* of the machine that is the NetWorker server.
online indexes	The databases located on the server that contain all the information pertaining to the client backups and backup volumes.
operator	The person who monitors the server status, loads backup volumes into the server devices, and otherwise executes day-to-day tasks using NetWorker.
override	A backup level that takes place instead of the scheduled one.
parallelism	A NetWorker feature that enables the NetWorker server to 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.
performance	Speed and efficiency with which an operation is performed.
pool	A feature that allows the NetWorker administrator to sort backup data to selected volumes. A volume pool contains a collection of backup volumes to which specific data has been backed up.
preconfigured	Existing selections or configurations for different NetWorker features.

print	Send data to a printer.
qic	A choice in the Devices window; represents quarter-inch cartridge tape.
recover	The NetWorker command used to browse the server index and recover files from a backup volume to a client's disk.
recovery	A method that recreates an image of the client filesystems and database on the NetWorker server machine.
recycle	A volume whose data has passed both its browse and retention policies and is available for relabeling.
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.
resource	An entity that desribes a NetWorker server state. In the NetWorker administration program, resources are represented as windows or dialog boxes that contain attributes, which you can use to view or set features of NetWorker. *See also*, attribute.
retention policy	A policy that determines how long save set entries are retained in the NetWorker server's media database.
retrieval	The process of locating and copying back files and directories that NetWorker has archived.
save	The NetWorker command that backs up client files to backup volumes and makes data entries in the online index.
save set	A group of files or a filesystem from a single client machine backed up onto storage media.

save set ID	An internal identification number that NetWorker assigns to a save set.
save set recover	The recovery of specified save sets to the NetWorker server.
save set status	An indication of whether a given save set is restorable, recoverable, or recyclable. The save set status also indicates whether or not 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.
scanner	The NetWorker command used to read a backup volume when the online indexes are no longer available.
server	The machine that runs the NetWorker software, contains the online indexes, and provides backup and recover services to clients on a network.
service	A program that is not invoked explicitly, but lies dormant waiting for a specified condition(s) to occur.
shell prompt	A cue for input in a shell window where you enter a command. *See also* command-line interface.
silo	A storage peripheral that can hold hundreds or thousands of volumes. Silo volumes are identified by barcodes, not by slot numbers. The robotics in a silo are controlled by silo management software on a silo server machine.
skip (s)	A backup level in which files are skipped and not backed up.
standalone device	A storage device that contains a single drive for backing up data.
storage device	The hardware that reads and writes data during backup, recover, or other NetWorker operations.

system administrator	The person normally responsible for installing, configuring, and maintaining NetWorker.
user	The person who can use NetWorker from his or her workstation to back up and recover files.
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.
volume ID	The internal identification assigned to a backup volume by NetWorker.
volume name	The name you assign to a backup volume when it is labeled.
volume pool	A feature that allows 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.

Index

E

F

H

I

L

M

R

S

T

U

V

W