



EMC[®] NetWorker[®]

Version 9.0.x

Performance Optimization Planning Guide

302-001-774

REV 04

EMC²

Copyright © 2000-2016 EMC Corporation. All rights reserved. Published in the USA.

Published June 2016

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

The information in this publication is provided as is. EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose. Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

EMC², EMC, and the EMC logo are registered trademarks or trademarks of EMC Corporation in the United States and other countries. All other trademarks used herein are the property of their respective owners.

For the most up-to-date regulatory document for your product line, go to EMC Online Support (<https://support.emc.com>).

EMC Corporation
Hopkinton, Massachusetts 01748-9103
1-508-435-1000 In North America 1-866-464-7381
www.EMC.com

CONTENTS

Figures		5
Tables		7
Preface		9
Chapter 1	Overview	13
	Introduction.....	14
	NetWorker data flow.....	14
Chapter 2	Size the NetWorker Environment	17
	Expectations.....	18
	Determine backup environment performance expectations.....	18
	Determining the backup window.....	18
	Determining the required backup expectations.....	18
	System components.....	19
	System.....	19
	Memory requirements for the NetWorker Server and NMC.....	20
	System bus requirements.....	22
	Storage considerations.....	23
	Storage IOPS requirements.....	24
	NetWorker server and storage node disk write latency.....	25
	Storage performance recommendations.....	26
	Backup operation requirements.....	27
	NetWorker kernel parameter requirements.....	28
	Parallel save stream considerations.....	28
	NetWorker resource considerations.....	35
	Wrapper script to pass environment variable to save and savegroup commands.....	35
	Internal maintenance task requirements.....	36
	Network.....	40
	Target device.....	40
	The component 70 percent rule.....	41
	Components of a NetWorker environment.....	41
	Datazone.....	42
	NetWorker Management Console.....	42
	Console database.....	43
	NetWorker server.....	44
	NetWorker storage node.....	45
	NetWorker client.....	46
	NetWorker databases.....	46
	Optional NetWorker Application Modules.....	47
	Virtual environments.....	47
	NetWorker deduplication nodes.....	47
	Recovery performance factors.....	47
	Connectivity and bottlenecks.....	47
	NetWorker database bottlenecks.....	52

Chapter 3	Tune Settings	55
	Optimize NetWorker parallelism.....	56
	Server parallelism.....	56
	Client parallelism.....	56
	Group parallelism.....	56
	Multiplexing.....	57
	File system density.....	57
	Disk optimization.....	58
	Device performance tuning methods.....	58
	Input/output transfer rate.....	58
	Built-in compression.....	59
	Drive streaming.....	59
	Device load balancing.....	59
	Fragmenting a disk drive.....	59
	Network devices.....	60
	Fibre Channel latency.....	60
	Data Domain.....	61
	AFTD device target and max sessions.....	62
	Number of virtual device drives versus physical device drives.....	63
	Network optimization.....	64
	Advanced configuration optimization.....	64
	Operating system TCP stack optimization.....	64
	Advanced tuning.....	65
	Network latency.....	65
	Ethernet duplexing.....	66
	Firewalls.....	67
	Jumbo frames.....	67
	Congestion notification.....	67
	TCP buffers.....	68
	Increase TCP backlog buffer size.....	69
	NetWorker socket buffer size.....	70
	IRQ balancing and CPU affinity.....	70
	Interrupt moderation.....	71
	TCP chimney offloading.....	71
	Name resolution.....	72
Chapter 4	Test Performance	73
	Determine symptoms.....	74
	Monitor performance.....	74
	Determining bottlenecks by using a generic FTP test.....	75
	Testing setup performance using the dd test.....	75
	Test disk performance by using bigasm and uasm.....	75
	The bigasm directive.....	75
	The uasm directive.....	76
	TCP window size and network latency considerations.....	76

FIGURES

1	NetWorker backup data flow.....	14
2	NetWorker recover data flow.....	15
3	PSS performance gains between NetWorker 8.2 and NetWorker 9.0.1 releases.....	29
4	NetWorker datazone components.....	42
5	Network device bottleneck.....	48
6	Updated network.....	49
7	Updated client.....	50
8	Dedicated SAN.....	51
9	Raid array.....	52
10	NetWorker server write throughput degradation.....	53
11	Files versus throughput.....	58
12	Fibre Channel latency impact on data throughput.....	61
13	Network latency on 10/100 MB per second.....	66
14	Network latency on 1 GB.....	66
15	Large Density FS Backup - WAN between NetWorker Clients and Server (High Packet Loss)	77
16	Large Density FS Backup - WAN between NetWorker Clients and Server (Low Packet Loss)	77
17	High Density FS Backup Performance - WAN between NetWorker Clients and Server	78
18	Large Density FS Backup Performance - WAN between NetWorker Clients and Data Domain	79
19	High Density FS Backup Performance - WAN between NetWorker Clients and Data Domain	79
20	Large Density FS Clone Performance - WAN between NetWorker Clients and Data Domain	80
21	High Density FS Clone Performance - WAN between NetWorker Clients and Data Domain	80
22	Large Density FS Recovery Performance - WAN between NetWorker Clients and Data Domain	81
23	High Density FS Recovery Performance - WAN between NetWorker Clients and Data Domain.....	81

FIGURES

TABLES

1	Revision history.....	9
2	Style conventions.....	11
3	Memory requirements based on size of configuration	21
4	Bus specifications	23
5	Disk write latency results and recommendations	25
6	PSS support by NetWorker release.....	29
7	Required IOPS for NetWorker server operations	37
8	Disk drive IOPS values	39
9	The effect of blocksize on an LTO-4 tape drive	61
10	Tolerable Range for Low Density file system.....	82
11	Tolerable Range for High Density file system.....	82

Preface

As part of an effort to improve its product lines, EMC periodically releases revisions of its software and hardware. Therefore, some functions that are described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features.

Contact your EMC technical support professional if a product does not function correctly or does not function as described in this document.

Note

This document was accurate at publication time. Go to EMC Online Support (<https://support.emc.com>) to ensure that you are using the latest version of this document.

Purpose

This document describes how to size and optimize the NetWorker software.

Audience

This document is intended for the NetWorker software administrator.

Revision history

The following table presents the revision history of this document.

Table 1 Revision history

Revision	Date	Description
01	September 24, 2015	First release of this document for EMC NetWorker 9.0
02	October 23, 2015	Added references to Windows Server 2012 in the "Network optimization" section of the "Tune Settings" chapter
03	April 15, 2016	Updated Memory requirements based on size of configuration in the section "Memory requirements for the NetWorker Server and NMC"
04	June 29, 2016	First release of this document for EMC NetWorker 9.0.1

Related documentation

The NetWorker documentation set includes the following publications, available on EMC Online Support:

- *EMC NetWorker Online Software Compatibility Guide*
Provides a list of client, server, and storage node operating systems supported by the EMC information protection software versions. You can access the Online Software Compatibility Guide on the EMC Online Support site at <https://support.emc.com>. From the Support by Product pages, search for NetWorker using "Find a Product", and then select the Install, License, and Configure link.
- *EMC NetWorker Administration Guide*
Describes how to configure and maintain the NetWorker software.
- *EMC NetWorker Network Data Management Protocol (NDMP) User Guide*

Describes how to use the NetWorker software to provide data protection for NDMP filers.

- *EMC NetWorker Cluster Integration Guide*
Contains information related to configuring NetWorker software on cluster servers and clients.
- *EMC NetWorker Installation Guide*
Provides information on how to install, uninstall, and update the NetWorker software for clients, storage nodes, and servers on all supported operating systems.
- *EMC NetWorker Updating from a Previous Release Guide*
Describes how to update the NetWorker software from a previously installed release.
- *EMC NetWorker Release Notes*
Contains information on new features and changes, fixed problems, known limitations, environment and system requirements for the latest NetWorker software release.
- *EMC NetWorker Command Reference Guide*
Provides reference information for NetWorker commands and options.
- *EMC NetWorker Data Domain Boost Integration Guide*
Provides planning and configuration information on the use of Data Domain devices for data deduplication backup and storage in a NetWorker environment.
- *EMC NetWorker Performance Optimization Planning Guide*
Contains basic performance tuning information for NetWorker.
- *EMC NetWorker Server Disaster Recovery and Availability Best Practices Guide*
Describes how to design and plan for a NetWorker disaster recovery. However, it does not provide detailed disaster recovery instructions. The Disaster Recovery section of the NetWorker Procedure Generator (NPG) provides step-by-step disaster recovery instructions.
- *EMC NetWorker Snapshot Management Integration Guide*
Describes the ability to catalog and manage snapshot copies of production data that are created by using mirror technologies on EMC storage arrays.
- *EMC NetWorker Snapshot Management for NAS Devices Integration Guide*
Describes how to catalog and manage snapshot copies of production data that are created by using replication technologies on NAS devices.
- *EMC NetWorker VMware Integration Guide*
Provides planning and configuration information on the use of VMware in a NetWorker environment.
- *EMC NetWorker Error Message Guide*
Provides information on common NetWorker error messages.
- *EMC NetWorker Licensing Guide*
Provides information about licensing NetWorker products and features.
- *EMC NetWorker REST API Getting Started Guide*
Describes how to configure and use the NetWorker REST API to create programmatic interfaces to the NetWorker server.
- *EMC NetWorker REST API Reference Guide*
Provides the NetWorker REST API specification used to create programmatic interfaces to the NetWorker server.
- **EMC NetWorker Management Console Online Help**
Describes the day-to-day administration tasks performed in the NetWorker Management Console and the NetWorker Administration window. To view the online help, click **Help** in the main menu.

- **EMC NetWorker User Online Help**
The NetWorker User program is the Windows client interface. Describes how to use the NetWorker User program which is the Windows client interface connect to a NetWorker server to back up, recover, archive, and retrieve files over a network.

Special notice conventions that are used in this document

EMC uses the following conventions for special notices:

NOTICE

Identifies content that warns of potential business or data loss.

Note

Contains information that is incidental, but not essential, to the topic.

Typographical conventions

EMC uses the following type style conventions in this document:

Table 2 Style conventions

Bold	Used for names of interface elements, such as names of buttons, fields, tab names, and menu paths (what the user specifically selects or clicks)
<i>Italic</i>	Used for full titles of publications that are referenced in text
Monospace	Used for: <ul style="list-style-type: none"> • System code • System output, such as an error message or script • Pathnames, file names, prompts, and syntax • Commands and options
<i>Monospace italic</i>	Used for variables
Monospace bold	Used for user input
[]	Square brackets enclose optional values
	Vertical bar indicates alternate selections - the bar means “or”
{ }	Braces enclose content that the user must specify, such as x or y or z
...	Ellipses indicate non-essential information that is omitted from the example

Where to get help

EMC support, product, and licensing information can be obtained as follows:

Product information

For documentation, release notes, software updates, or information about EMC products, go to EMC Online Support at <https://support.emc.com>.

Technical support

Go to EMC Online Support and click Service Center. Several options for contacting EMC Technical Support appear on the site. Note that to open a service request, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or with questions about your account.

Online communities

Go to EMC Community Network at <https://community.emc.com> for peer contacts, conversations, and content on product support and solutions. Interactively engage online with customers, partners, and certified professionals for all EMC products.

Your comments

Your suggestions help to improve the accuracy, organization, and overall quality of the user publications. Send your opinions of this document to DPAD.Doc.Feedback@emc.com.

CHAPTER 1

Overview

- [Introduction](#)..... 14
- [NetWorker data flow](#)..... 14

Introduction

The EMC® NetWorker® software is a network storage management application that is optimized for the high-speed backup and recovery operations of large amounts of complex data across entire datazones. This guide addresses non-disruptive performance tuning options. Although some physical devices may not meet the expected performance, it is understood that when a physical component is replaced with a better performing device, another component becomes a bottle neck. This manual attempts to address NetWorker performance tuning with minimal disruptions to the existing environment. It attempts to fine-tune feature functions to achieve better performance with the same set of hardware, and to assist administrators to:

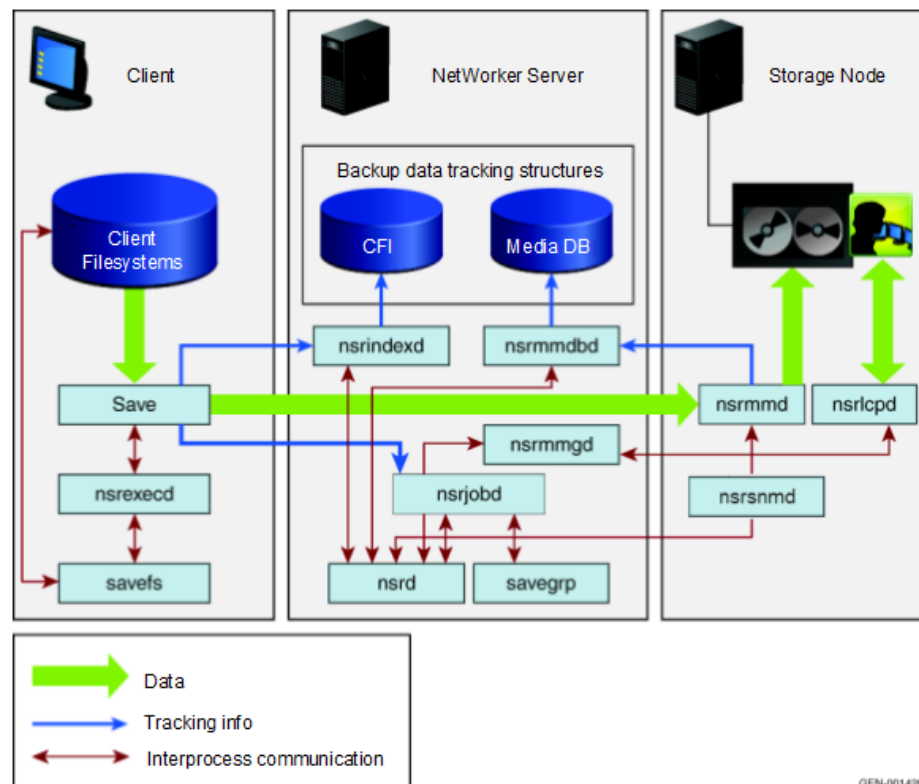
- Understand data transfer fundamentals
- Determine requirements
- Identify bottlenecks
- Optimize and tune NetWorker performance

NetWorker data flow

The following figures illustrate the backup and recover data flow for components in a NetWorker datazone.

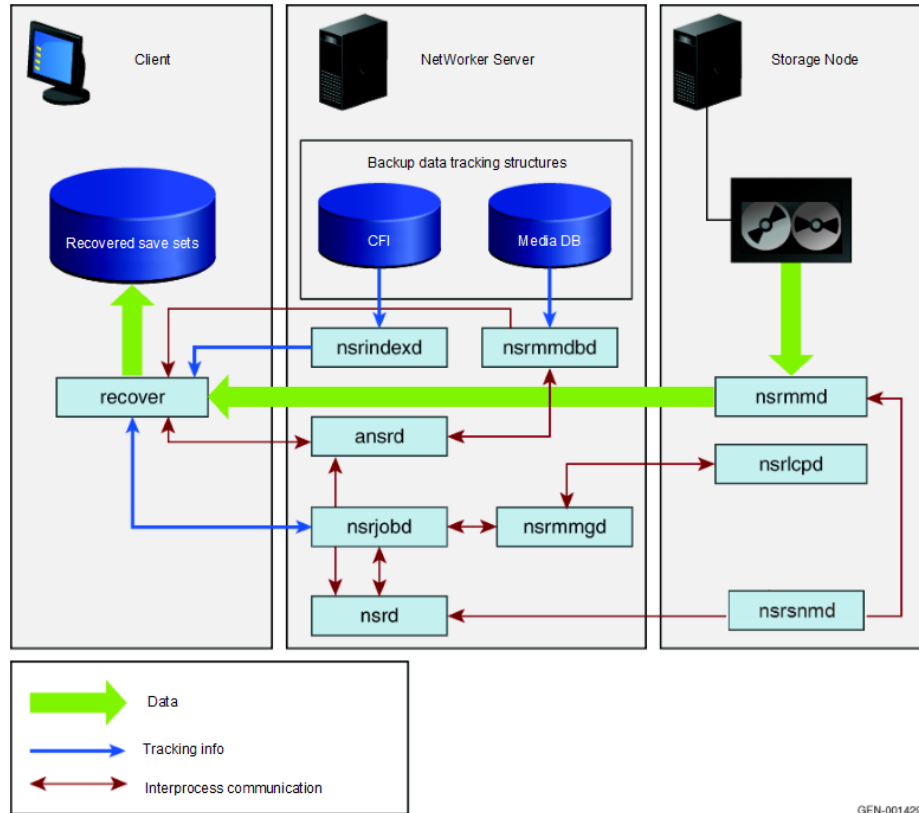
The following figures are simplified diagrams, and not all interprocess communication is shown. There are many other possible backups and recover data flow configurations.

Figure 1 NetWorker backup data flow



GEN-001428

Figure 2 NetWorker recover data flow



GEN-001429

CHAPTER 2

Size the NetWorker Environment

This chapter describes how to best determine backup and system requirements. The first step is to understand the environment. Performance issues are often attributed to hardware or environmental issues. An understanding of the entire backup data flow is important to determine the optimal performance expected from the NetWorker software.

- [Expectations](#)..... 18
- [System components](#)..... 19
- [Storage considerations](#)..... 23
- [Backup operation requirements](#)..... 27
- [Components of a NetWorker environment](#)..... 41
- [Recovery performance factors](#)..... 47
- [Connectivity and bottlenecks](#)..... 47

Expectations

You can determine the backup performance expectations and required backup configurations for your environment based on the Recovery Time Objective (RTO) for each client.

Determine backup environment performance expectations

It is important to determine performance expectations, while keeping in mind the environment and the devices used.

Sizing considerations for the backup environment are listed here:

- Review the network and storage infrastructure information before setting performance expectations for your backup environment including the NetWorker server, storage nodes, and clients.
 - Review and set the RTO for each client.
 - Determine the backup window for each NetWorker client.
 - List the amount of data to be backed up for each client during full and incremental backups.
 - Determine the data growth rate for each client.
 - Determine client browse and retention policy requirements.
- Some suggestions to help identify bottlenecks and define expectations are:

- Create a diagram
- List all system, storage, network, and target device components
- List data paths
- Mark down the bottleneck component in the data path of each client

[Connectivity and bottlenecks on page 47](#) provides examples of possible bottlenecks in the NetWorker environment.

Determining the backup window

It is very important to know how much down time is acceptable for each NetWorker client. This dictates the recovery time objective (RTO). Review and document the RTO for each NetWorker client to determine the backup window for each client.

Procedure

1. Verify the available backup window for each NetWorker client.
2. List the amount of data that must be backed up from the clients for full or incremental backups.
3. List the average daily/weekly/monthly data growth on each NetWorker client.

Determining the required backup expectations

Often it is not possible to construct a backup image from a full backup and multiple incremental backups if the acceptable down time is limited. Full backups might be required more frequently which results in a longer backup window. This also increases network bandwidth requirements.

Methods to determine the required backup configuration expectations for the environment are listed here:

- Verify the existing backup policies and ensure that the policies will meet the RTO for each client.
- Estimate backup window for each NetWorker client based on the information collected.
- Determine the organization of the separate NetWorker client groups based on these parameters:
 - Backup window
 - Business criticality
 - Physical location
 - Retention policy
- Ensure that RTO can be met with the backup created for each client. Backups become increasingly expensive as the acceptable downtime/backup window decreases.

System components

Every backup environment has a bottleneck. It may be a fast bottleneck, but the bottleneck will determine the maximum throughput obtainable in the system. Backup and restore operations are only as fast as the slowest component in the backup chain.

Performance issues are often attributed to hardware devices in the datazone. This guide assumes that hardware devices are correctly installed and configured.

This section discusses how to determine requirements. For example:

- How much data must move?
- What is the backup window?
- How many drives are required?
- How many CPUs are required?

Devices on backup networks can be grouped into four component types. These are based on how and where devices are used. In a typical backup network, the following four components are present:

- System
- Storage
- Network
- Target device

System

Several system configuration components impact performance:

- CPU
- Memory
- System bus (this determines the maximum available I/O bandwidth)

CPU requirements

Determine the optimal number of CPUs required, if 5 MHz is required to move 1 MB of data from a source device to a target device. For example, a NetWorker server, or storage

node backing up to a local tape drive at a rate of 100 MB per second, requires 1 GHz of CPU power:

- 500 MHz is required to move data from the network to a NetWorker server or storage node.
- 500 MHz is required to move data from the NetWorker server or storage node to the backup target device.

NOTICE

1 GHz on one type of CPU does not directly compare to a 1 GHz of CPU from a different vendor.

The CPU load of a system is impacted by many additional factors. For example:

- High CPU load is not necessarily a direct result of insufficient CPU power, but can be a side effect of the configuration of the other system components.
- Drivers:
Be sure to investigate drivers from different vendors as performance varies. Drivers on the same operating system achieve the same throughput with a significant difference in the amount of CPU used.
- Disk drive performance:
 - On a backup server with 400 or more clients in `/nsr`, a heavily used disk drive often results in CPU use of more than 60 percent. The same backup server in `/nsr` on a disk array with low utilization, results in CPU use of less than 15 percent.
 - On UNIX, and Windows if much CPU time is spent in privileged mode or if a percentage of CPU load is higher in system time than user time, it often indicates that the NetWorker processes are waiting for I/O completion. If the NetWorker processes are waiting for I/O, the bottleneck is not the CPU, but the storage that is used to host NetWorker server.
 - On Windows, if much time is spent on Deferred Procedure Calls it often indicates a problem with device drivers.
- Monitor CPU use according to the following classifications:
 - User mode
 - System mode
- Hardware component interrupts cause high system CPU use resulting poor performance. If the number of device interrupts exceed 10,000 per second, check the device.

Memory requirements for the NetWorker Server and NMC

The NetWorker 9.0 software can consume a large amount of memory for the NetWorker server and NMC.

Also, the following services introduced in NetWorker 9.0 can be memory-intensive since these services use Java and Apache tomcat:

- Web-based authentication service
- Message Queue adapter
- Hyper-V FLR in NMM

NetWorker 9.0 requires an additional 2 GB, 4 GB and 16 GB memory for small, medium and large scale configurations respectively due to the addition of these features. Also, as

a best practice EMC recommends that Java heap memory should typically consume no more than 25% of physical memory.

Note

EMC recommends that you avoid installing NMC on the NetWorker server. If you install NMC on the NetWorker server, then the memory requirements for the NetWorker server identified in the following tables will be in addition to the minimum NMC memory requirements specified in the section [Minimum system requirements for the NMC server on page 42](#).

The following table lists the minimum memory requirements for the NetWorker Server.

Table 3 Memory requirements based on size of configuration

Size of configuration	Minimum CPU required	Minimum memory required
Small configuration (maximum 500 clients)	1 CPU, 3.0 GHz	8 GB RAM
Medium configuration (maximum 1000 clients)	1 CPU, 3.4 GHz	16 GB RAM
Large configuration (maximum 2000 clients)	2 CPU, 3.4 GHz	32 GB RAM

1024 parallelism value

Based on performance observations for savegroups similar to the following configuration, the minimum required memory is 1 GB on Linux, 600-700 MB on Solaris, and 250-300 MB on Windows:

- 25 clients
- Client parallelism = 8
- 5 remote storage nodes with 32 AFTD devices
- Default device Target sessions (4)
- Default device Max sessions (32)

NOTICE

Increasing the server parallelism value affects the NetWorker server IOPS on the index and media databases.

Monitor the pagefile or swap use

Memory paging should not occur on a dedicated backup server as it will have a negative impact on performance in the backup environment.

Client Direct attribute for direct file access (DFA)

There are many conditions to consider when enabling DFA by using the Client Direct attribute.

The following are the considerations for enabling DFA by using the Client Direct attribute:

- Ensure there is enough CPU power on the client to take advantage of DFA-DD increased performance capability. In most cases, Client Direct significantly improves

backup performance. The DFA-DD backup requires approximately 2-10% more CPU load for each concurrent session.

- Each save session using DFA-DD requires up to 70 MB of memory. If there are 10 DFA streams running, then the memory required on a client for all DFA sessions is 700 MB.
- Save sessions to DFA-AFTD use less memory and CPU cycles as compared to backup running to DFA-DD using Boost. Save sessions using DFA-AFTD use only slightly more memory and CPU cycles as compared to traditional saves with mmd.

System bus requirements

Although HBA/NIC placement are critical, the internal bus is probably the most important component of the operating system. The internal bus provides communication between internal computer components, such as CPU, memory, disk, and network.

Bus performance criteria

Bus performance depends on several factors:

- Type of bus
- Data width
- Clock rate
- Motherboard

System bus considerations

There are considerations to note that concern the bus performance.

- A faster bus does not guarantee faster performance
- Higher end systems have multiple buses to enhance performance
- The bus is often the main bottleneck in a system

System bus recommendations

It is recommended to use PCIeExpress for both servers and storage nodes to reduce the chance for I/O bottlenecks.

NOTICE

Avoid using old bus types or high speed components optimized for old bus type as they generate too many interrupts causing CPU spikes during data transfers.

PCI-X and PCIeExpress considerations

There are considerations that specifically concern the PCI-X and PCIeExpress buses.

- PCI-X is a half-duplex bi-directional 64-bit parallel bus.
- PCI-X bus speed may be limited to the slowest device on the bus, be careful with card placement.
- PCIeExpress is full-duplex bi-directional serial bus using 8/10 encoding.
- PCIeExpress bus speed may be determined per each device.
- Do not connect a fast HBA/NIC to a slow bus, always consider bus requirements. Silent packet drops can occur on a PCI-X 1.0 10GbE NIC, and bus requirements cannot be met.
- Hardware that connects fast storage to a slower HBA/NIC will slow overall performance.

The [component 70 percent rule on page 41](#) provides details on the ideal component performance levels.

Bus speed requirements

Required bus speeds are based on the Fibre Channel size:

- 4 GB Fibre Channel requires 425 MB/s
- 8 GB Fibre Channel requires 850 MB/s
- 10 GB Fibre Channel requires 1,250 MB/s

Bus specifications

Bus specifications are based on bus type, MHz, and MB per second.

Bus specifications for specific buses are listed in the following table.

Table 4 Bus specifications

Bus type	MHz	MB/second
PCI 32-bit	33	133
PCI 64-bit	33	266
PCI 32-bit	66	266
PCI 64-bit	66	533
PCI 64-bit	100	800
PCI-X 1.0	133	1,067
PCI-X 2.0	266	2,134
PCI-X 2.0	533	4,268
PCleXpress 1.0 x 1		250
PCleXpress 1.0 x 2		500
PCleXpress 1.0 x 4		1,000
PCleXpress 1.0 x 8		2,000
PCleXpress 1.0 x 16		4,000
PCleXpress 1.0 x 32		8,000
PCleXpress 2.0 x 8		4,000
PCleXpress 2.0 x 16		8,000
PCleXpress 2.0 x 32		16,000

Storage considerations

There are components that impact the performance of storage configurations:

- Storage connectivity:
 - Local versus SAN attached versus NAS attached.
 - Use of storage snapshots.
The type of snapshot technology used determines the read performance.

- **Storage replication:**
Some replication technologies add significant latency to write access which slows down storage access.
- **Storage type:**
 - Serial ATA (SATA) computer bus is a storage-interface for connecting host bus adapters to storage devices such as hard disk drives and optical drives.
 - Fibre Channel (FC) is a gigabit-speed network technology primarily used for storage networking.
 - Flash is a non-volatile computer storage used for general storage and the transfer of data between computers and other digital products.
- **I/O transfer rate of storage:**
I/O transfer rate of storage is influenced by different RAID levels, where the best RAID level for the backup server is RAID1 or RAID5. Backup to disk should use RAID3.
- **Scheduled I/O:**
If the target system is scheduled to perform I/O intensive tasks at a specific time, schedule backups to run at a different time.
- **I/O data:**
 - Raw data access offers the highest level of performance, but does not logically sort saved data for future access.
 - File systems with a large number of files have degraded performance due to additional processing required by the file system.
- **Compression:**
If data is compressed on the disk, the operating system or an application, the data is decompressed before a backup. The CPU requires time to re-compress the files, and disk speed is negatively impacted.

Storage IOPS requirements

The file system that is used to host the NetWorker data (`/nsr`) must be a native file system that is supported by the operating system vendor for the underlying operating system and must be fully Posix compliant.

If the storage performance requirements measured in I/O operations per second (IOPS) documented in this section are not met, NetWorker server performance is degraded and can be unresponsive for short periods of time.

If storage performance falls below 50% of the preferred IOPS requirements:

- NetWorker server performance can become unreliable
- NetWorker server can experience prolonged unresponsive periods
- Backup jobs can fail
NetWorker server requirements, with respect to storage performance are determined by the following:
 - NetWorker datazone monitoring
 - Backup jobs
 - Maintenance tasks
 - Reporting tasks
 - Manual tasks

NetWorker server and storage node disk write latency

It is important to determine the requirements for the NetWorker server and the storage node write latency. Write latency for `/nsr` on NetWorker servers, and storage nodes is more critical for the storage hosting `/nsr` than is the overall bandwidth. This is because NetWorker uses a very large number of small random I/O for internal database access.

The following table lists the effects on performance for disk write latency during NetWorker backup operations.

Table 5 Disk write latency results and recommendations

Disk write latency in milliseconds (ms)	Effect on performance	Recommended
25 ms and below	<ul style="list-style-type: none"> Stable backup performance Optimal backup speeds 	Yes
50 ms	<ul style="list-style-type: none"> Slow backup performance (the NetWorker server is forced to throttle database updates) Delayed and failed NMC updates 	No
100 ms	Failed savegroups and sessions	No
150–200 ms	<ul style="list-style-type: none"> Delayed NetWorker daemon launch Unstable backup performance Unprepared volumes for write operations Unstable process communication 	No

NOTICE

Avoid using synchronous replication technologies or any other technology that adversely impacts latency.

Recommended server and storage node disk settings

It is important to consider recommendations for optimizing NetWorker server and storage node disk performance:

- For NetWorker servers under increased load (number of parallel sessions occurring during a backup exceeds 100 sessions), dedicate a fast disk device to host NetWorker databases.
- For disk storage configured for the NetWorker server, use RAID 10.
- For large NetWorker servers with server parallelism higher than 400 parallel sessions, split the file systems that are used by the NetWorker server. For example, split the `/nsr` folder from a single mount to multiple mount points for:
 - `/nsr`
 - `/nsr/res`
 - `/nsr/index`
 - `/nsr/mm`
- For NDMP backups, configure a separate location on the NetWorker server for the `/nsr/tmp` folder to accommodate large temporary file processing.

- Use the operating system to handle parallel file system I/O even if all mount points are on the same physical location. The operating system handles parallel file system I/O more efficiently than the NetWorker software.
- Use RAID 3 for disk storage for AFTD.
- For antivirus software, disable scanning of the NetWorker databases. If the antivirus software scans the `/nsr` folder, performance degradation, time-outs, or NetWorker database corruption can occur because of frequent file open/close requests. The antivirus exclude list should also include NetWorker storage node locations that are used for Advanced File Type Device (AFTD).

Note

Disabled antivirus scanning of specific locations might not be effective if it includes all locations during file access, despite the exclude list if it skips scanning previously accessed files. Contact the specific vendor to obtain an updated version of the antivirus software.

- For file caching, aggressive file system caching can cause commit issues for:
 - The NetWorker server: All NetWorker databases can be impacted (`nsr\res`, `nsr\index`, `nsr\mm`).
 - The NetWorker storage node: When configured to use Advanced File Type Device (AFTD).
Be sure to disable delayed write operations, and use driver Flush and Write-Through commands instead.
- Disk latency considerations for the NetWorker server are higher than for typical server applications as NetWorker uses committed I/O: Each write to the NetWorker internal database must be acknowledged and flushed before next write is tried. This setting avoids any potential data loss in internal databases.
Where storage is replicated or mirrored for `/nsr`, consider the following:
 - Do not use software based replication as it adds an additional layer to I/O throughput and causes unexpected NetWorker behavior.
 - With hardware based replication, the preferred method is asynchronous replication as it does not add latency on write operations.
 - Do not use synchronous replication over long distance links, or links with non-guaranteed latency.
 - SANs limit local replication to 12 km and longer distances require special handling.
 - Do not use TCP networks for synchronous replication as they do not guarantee latency.
 - Consider the number of hops as each hardware component adds latency.

Storage performance recommendations

The same physical storage sub-system can perform differently depending on the configuration. For example, splitting a single NetWorker mount point (`/nsr`) into multiple mount points can significantly increase performance due to the parallelism of the file system handler in the operating system.

The NetWorker software does not use direct I/O, but issues a sync request for each write operation. This setting ensures that data is flushed on the disk and avoids data loss if a system failure (committed I/O writes) occurs. Therefore write caching on the operating system has minimal, or no impact. However, hardware-based write-back cache can significantly improve NetWorker server performance.

Processes can be single threaded or multi-threaded, depending on process itself and whether it is configurable. Blocking-IO (MMDB, RAP) provides the best data protection. The exception is the indexDB where each client has its own I/O stream.

General recommendations for NetWorker server metadata storage are grouped depending on the NetWorker database type:

- The RAP database is file-based with full file read operations with an average I/O of > 1 KB.
- The MMDB is block-based with a fixed block size of 32 KB with many read operations and fewer write operations.
- Set separate mount points for each database on the Flash drive to avoid I/O bottlenecks on the NetWorker server update to the RAP, MMDB, and Index database.
- The indexDB is primarily based on sequential write operations with no fixed block size and few read operations. A lower storage tier such as SAS or SATA based storage is sufficient for the indexDB.
- The temporary NetWorker folder, `/nsr/tmp` is used heavily during index merge operations for NDMP backups. The temporary folder should reside on higher tier storage, such as FC drives.

I/O Pattern Considerations

The NetWorker I/O pattern for access to configuration and metadata databases varies depending on the database and its use. However, it generally includes certain elements:

- Normal backup operations: 80% write / 20% read
- Cross-check operations: 20% write / 80% read
- Reporting operations: 100% read

Based on these patterns, the daily cross-check should be performed outside of the primary backup window. Also, external solutions that provide reporting information should be configured to avoid creating excessive loads on the NetWorker metadata databases during the production backup window. I/O block size also varies depending on database and use-case, but generally it's rounded to 8 KB requests.

NetWorker datazone monitoring recommendations

Storage must provide a minimum of 30 IOPS to the NetWorker server. This number increases as the NetWorker server load increases.

Backup operation requirements

Requirements for starting and running backup operations is the largest portion of the NetWorker software workload:

- Depending on the load, add to the IOPS requirements the maximum concurrent sessions on the NetWorker server, and divide this number by three. The maximum NetWorker server parallelism is 1024, therefore the highest possible load is $1024/3=340$ IOPS.
- IOPS requirements increase if the NetWorker software must perform both index and bootstrap backups simultaneously. In this case, add:
 - 50 IOPS for small servers
 - 150 IOPS for medium servers
 - 400 IOPS for large servers

[Manual NetWorker server task requirements on page 37](#) provides guidelines for small, medium, and large NetWorker servers.

Note

Add the additional IOPS only if the bootstrap backup runs concurrently with the normal backup operations. If the bootstrap backup is configured to run when the NetWorker server is idle, the IOPS requirements do not increase.

- IOPS requirements increase if the NetWorker software is configured to start many jobs simultaneously.
To accommodate load spikes, add one IOPS for each parallel session that is started.
It is recommended not to start more than 40 clients per group with the default client parallelism of four. The result is 160 IOPS during group startup.
Starting many clients simultaneously can lead to I/O system starvation.
- Each volume request results in a short I/O burst of approximately 200 IOPS for a few seconds.
For environments running a small number of volumes the effect is minimal. However, for environments with frequent mount requests, a significant load is added to the NetWorker server. In this case, add 100 IOPS for high activity (more than 50 mount requests per hour). To avoid the excessive load, use a smaller number of large volumes.
- NDMP backups add additional load due to index post-processing.
For large NDMP environment backups with more than 10 million files, add an additional 120 IOPS.

NetWorker kernel parameter requirements

Create a separate startup script for the NetWorker servers with heavy loads by enabling the following environment variables before the NetWorker services start:

- *tcp_backlog_queue*: Add the appropriate kernel parameters in the startup script based on the operating system
- *Open file descriptors*: Change the open file descriptors parameter to a minimum of 8192 required on NetWorker servers with a heavy load

Note

Use the default startup script on the NetWorker storage nodes and clients. The *tcp_backlog_queue*, and the *open file descriptor* parameters are not required on storage nodes and clients.

Parallel save stream considerations

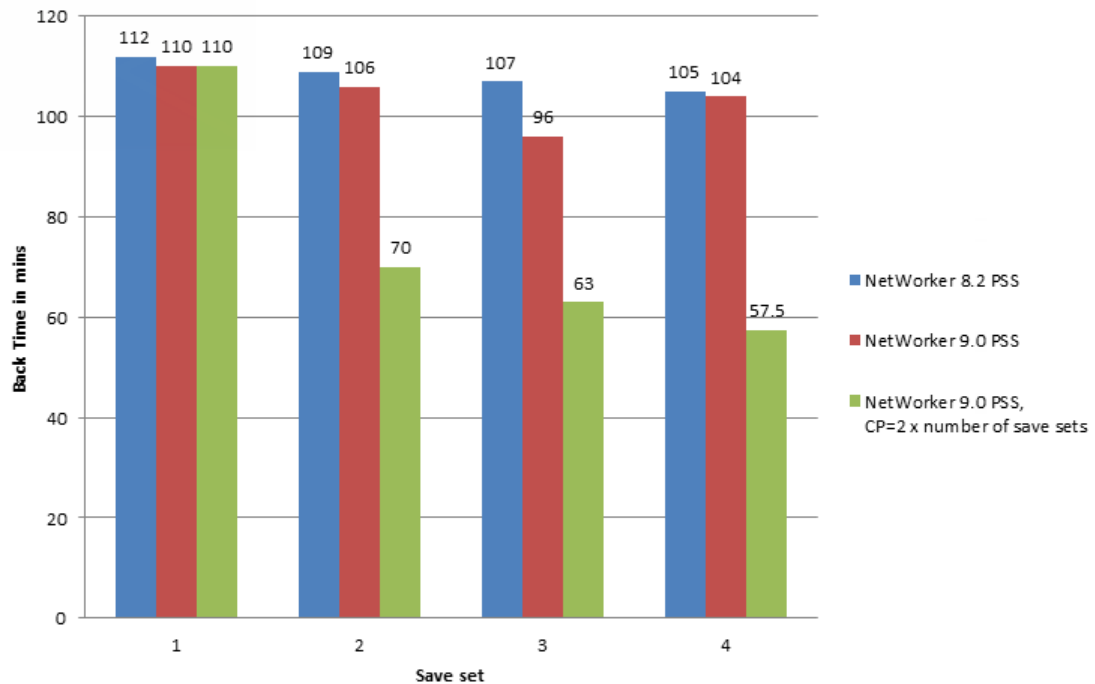
Enabling the parallel save streams (PSS) feature for a Client resource allows you to back up each save set for the client by using multiple parallel save streams to one or more destination backup devices. The save set entry is also called a save point, which is often a UNIX file system mount directory or a Windows volume drive letter. PSS is used for the scheduled, file-based backup of file systems. Significant parallel performance gains are possible during PSS backups and subsequent recoveries.

In NetWorker 8.1 and 8.2 releases, PSS does static division of the client parallelism (CP) up-front among the client's save points, also known as save sets, and starts all simultaneously. A sufficiently high client parallelism is therefore required.

In NetWorker 9.0.1, PSS schedules the save points at four parallel save streams each by default, and handles in batches according to the client parallelism value. A high CP value is therefore not required.

The following graph shows the potential performance gains between NetWorker 8.2 and NetWorker 9.0.1 releases due to the PSS enhancements. In this example, the NetWorker backup contains a total 2 TB of data that is divided among four save sets of different sizes.

Figure 3 PSS performance gains between NetWorker 8.2 and NetWorker 9.0.1 releases



Subsequent sections in this chapter break down the behavior of PSS in each release.

Parallel save streams considerations (previous to NetWorker 9.0)

Introduced in NetWorker 8.1, the parallel save streams (PSS) feature provides the ability for each Client resource save set entry to be backed up by multiple parallel save streams to one or more destination backup devices.

The following table lists the items that are supported or not supported for PSS in NetWorker 8.1 and NetWorker 8.2 releases.

Table 6 PSS support by NetWorker release

NetWorker release	Operating systems	Supported save sets	Supported backup types	Virtual and non-virtual synthetic full	Checkpoint restart
8.1 releases	UNIX, Linux	ALL, individual save points	Scheduled	No	No
8.2 and later	UNIX, Linux, Windows	ALL, individual save points including DISASTER	Scheduled	Yes	No

Table 6 PSS support by NetWorker release (continued)

NetWorker release	Operating systems	Supported save sets	Supported backup types	Virtual and non-virtual synthetic full	Checkpoint restart
		RECOVERY: \, deduplicated, and CSV volumes (Windows only)			

When a PSS enabled UNIX Client resource's parallelism value is greater than the resource's number of save points, the scheduled backup savegroup process divides the parallelism among the save points and starts PSS save processes for all the save points at approximately the same time. However, this is done within the limits of the following:

- The NetWorker server
- Group parallelism controls
- Media device session availability

EMC recommends setting the Client resource PSS parallelism value to two times or more the number of save points.

The number of streams for each PSS save point is determined before the backup from its client parallelism value and it remains fixed throughout the backup. It is a value 1–4 (maximum), where one indicates a single stream with a separate PSS process that traverses the save point's file system to determine the files to back up. The separation of processes for streaming data and traversing the file system can improve performance. Also, the number of save processes that run during a PSS save point backup is equal to the number of save stream processes that are assigned with two additional save processes for both the director and file system traversal processes.

The default maximum number of four streams for each PSS save point can be modified to a value 1–8 inclusive by setting the *NSR_SG_PSS_MAX_SP_SPLIT* environment variable in both UNIX and Windows. After setting the environment variable, restart the NetWorker services for the changes to take effect. Increasing the default maximum value can improve the performance for clients with very fast disks.

When the client parallelism is less than its number of save points, some save point backups are run in PSS mode, with only a single stream. Other save points are run in the default mode (non-PSS). Therefore, for consistent use of PSS, set the client parallelism to two times or more the number of save points. This ensures multiple streams for each save point.

It is recommended that large, fast file systems that should benefit from PSS be put in a new separate PSS-enabled Client resource that is scheduled separately from the client's other save points. Separate scheduling is achieved by using two different save groups with different runtimes, but the same savegroup can be used if you avoid client disk parallel read contention. Also, use caution when enabling PSS on a single Client resource with the keyword **A11**. **A11** typically expands to include multiple small operating file systems that reside on the same installation disk. These file systems usually do not benefit from PSS but instead might waste valuable PSS multi-streaming resources.

Based on the second example, the */sp1* save set record is referred to as the master and its save set time is used in browsing and time-based recover operations. It references the two related records (dependents) through the ***mbs dependents** attribute. This attribute lists the portable long-format save set IDs of the dependents. Each dependent indirectly references its master through save set name and save time associations. Its master is the

save set record with the next highest save time and save set name with no prefix. Also, each master record has an ***mbs anchor save set time** attribute, which references its dependent with the earliest save set time.

PSS improves on manually dividing save point /sp1, into multiple sub-directories, /sp1/subdirA, /sp1/subdirB... and typing each subdirectory separately in the Client resource. PSS eliminates the need to do this and automatically performs better load balancing optimization at the file-level, rather than at the directory level that is used in the manual approach. PSS creates pseudo sub-directories corresponding to the media save set record names, for example, /sp1,<1>/sp1, and <2>/sp1.

Both time-based recovery and savegroup cloning automatically aggregate the multiple physical save sets of a save point PSS backup. The multiple physical dependent save sets remain hidden. However, there is no automatic aggregation in save set based recovery, scanner, nsrmm, or nsrclone -S manual command line usage. The -S option requires the PSS save set IDs of both master and dependents to be specified at the command line. However, the -S option should rarely be required with PSS.

When the following PSS client configuration settings are changed, the number of save streams can change for the next save point incremental backup:

- The number of save points
- The parallelism value

NetWorker automatically detects differences in the number of save streams and resets the backup to a level Full accordingly. This starts a new <full, incr, incr, ...> sequence of backups with the same number of media database save sets for each PSS save point backup.

This applies to non-full level numbers 1–9 in addition to incremental, which is also known as level 10.

NOTICE

The PSS incremental backup of a save point with zero to few files changed since its prior backup results in one or more empty media databases save sets (actual size of 4 bytes), which is to be expected.

Example 1

The following provides performance configuration alternatives for a PSS enabled client with the following backup requirements and constraints.

- Two savepoints: /sp200GB and /sp2000GB
- Save streams able to back up at 100 GB/hr
- Client parallelism is set to four (No more than four concurrent streams to avoid disk IO contention)

Based on these requirements and constraints, the following are specific configuration alternatives with the overall backup time in hours:

- A non-PSS Client resource with both savepoints at one stream each: 20 hours
- A single PSS Client resource with both /sp200GB at two streams and /sp2000GB at two streams for the same save group: 10 hours
- A non-PSS Client resource with /sp200GB at one stream and a PSS Client resource with /sp2000GB at three streams for the same client host and same save group: 6.7 hours

- A PSS Client resource with /sp200GB at four streams and another PSS Client resource with /sp2000GB at four streams for the same client but different sequentially scheduled save groups: 5.5 hours aggregate

Example 2

With client parallelism set to eight and three save points /sp1, /sp2, and /sp3 explicitly listed or expanded by the keyword **ALL** for UNIX, the number of PSS streams for each savepoint backup is three, three, and two respectively. The number of `mminfo` media database save set records is also three, three, and two respectively.

For a particular save point, /sp1, `mminfo`, and NMC save set query results shows three save set records each named /sp1, <1>/sp1, and <2>/sp1. These related records have unique save times that are close to one another. The /sp1 record always has the latest save time, that is, maximum save time, as it starts last. This makes time-based recovery aggregation for the entire save point /sp1 work automatically.

Example 3

For a PSS Windows save point backup, the number of streams per save point is estimated in the following two scenarios.

- The client parallelism per save point, where client parallelism=5, and the number of save points=2, the number of PSS streams is three for the first save point, and two streams for the second.
For the save set **ALL**, with two volumes and client parallelism=5, each volume (save point) gets two streams.
- Using client parallelism=4, every save point is given two save streams. Both `DISASTER_RECOVERY:\` volumes, `C:\`, and `D:\` are given two streams also.
For the save set **ALL**, the `DISASTER_RECOVERY:\` save set is considered to be a single save point. For this example, the system has `C:\`, `D:\`, and `E:\`, where `C:\`, and `D:\` are the critical volumes that make up the `DISASTER_RECOVERY:\` save set.

The save operation controls how the save points are started, and the total number of streams never exceeds the client parallelism value of 4.

Parallel save stream considerations in NetWorker 9.0.1

NetWorker 9.0.1 PSS provides potentially greater performance gains over earlier NetWorker releases due to the new save set aggregation approach.

Save set aggregation is where the NetWorker server starts a single save process per PSS-enabled client, with all client save sets passed to the single process for various processing optimizations, such as minimal Windows VSS snapshots and support for the following:

- Four parallel streams started per save set, subject to any client parallelism limitations that might prevent all save sets from starting at the same time.
- The ability to modify the number of parallel streams per save set by defining the new **PSS:streams_per_ss** option in the selected client resource's **save operations** attribute.
- Automatic stream reclaiming, which dynamically increases the number of active streams for an already running save set backup to maximize utilization of limited client parallelism conditions, also known as Dynamic PSS.

PSS is for clients with supported UNIX, Linux, and Windows operating systems. Supported save sets for PSS include **ALL**, and individual save points including

`DISASTER_RECOVERY`: \, deduplicated, and CSV volumes (Windows only). Checkpoint restart is not supported when you use PSS.

Unlike NetWorker releases previous to NetWorker 9.0.1, where PSS requires a sufficiently high CP value due to the upfront static division of CP among the client's save points which all get started at the same time, NetWorker 9.0.1 PSS schedules the save points at four parallel save streams each by default, and handles in batches according to the client parallelism value. Therefore, you do not require a high value for CP.

The default four parallel save streams per save point should be sufficient for most save points, however you can change the default value by setting the `PSS:streams_per_ss` environment variable in the client's save operations attribute.

On UNIX for example, launch NMC and the **NetWorker Administration** window, and then go to **View** > **Diagnostic Mode** > **Protection** > **Clients** > **Client Properties** > **Apps & Modules** > **Save operations** and set the following:

```
PSS:streams_per_ss=1,* , 2,/data[1-3] , 8,/data[4-5]
```

This setting will use one stream per client save set entry by default, with the exception of two streams for each of `/data1`, `/data2` and `/data3`, and eight streams for each of `/data4` and `/data5`. Client-supported wildcard characters can be used. After setting the environment variable, restart the NetWorker services for the changes to take effect. Increasing the default maximum value can improve the performance for clients with very fast disks.

On Windows, launch NMC and the **NetWorker Administration** window, and then go to **View** > **Diagnostic Mode** > **Protection** > **Clients** > **Client Properties** > **Apps & Modules** > **Save operations** and set the following:

```
PSS:streams_per_ss=2,C:\ , D:\ , 8 , E:\ , F:\HR
```

This Windows PSS client setting will continue to use the default four streams for each save point not explicitly listed here, but two streams each for the `C:\` and `D:\` drives, and eight streams each for the `E:\` drive and `F:\HR` folder.

Note

Note PSS backups currently ignore the policy workflow action's parallelism, previously known as the `savegrp` parallelism.

When you set the client parallelism to a value less than the number of save points, some save point backups run in PSS mode, with only a single stream, and other save points will run in the default mode (non-PSS). Therefore, for consistent use of PSS, maintain the default setting or set the client parallelism to the number of save points. This ensures multiple streams for each save point.

NOTICE

The PSS incremental backup of a save point with zero to few files changed since its prior backup will result in one or more empty media database save sets (actual size of 4 bytes), which is to be expected.

Example 1 PSS enabled, CP=6 with 3 client save points

In NetWorker releases previous to NetWorker 9.0.1, if you set CP=6 and have three client save points, PSS will start all save points together at two streams each, and each save point will remain at two streams, with each stream actively backing up files from the start.

NetWorker 9.0.1, however, would start save point one with four active backup streams, and simultaneously start save point two with two active streams and two idle streams. If save point one finishes first, then save point two could end up with four active streams, and save point three would then start with two active streams and two idle streams. Depending on the time it takes the save point to complete, save point two could remain as it started and save point three may start similar to how save point one started, with four active streams. An idle stream is one that has not yet started saving data and will only become active when CP allows. The total number of active streams from all save sets at any one point in time will not exceed CP. EMC recommends a value of 4 or a multiple of four to avoid idle streams.

Command line examples

Consider the command line examples for after a PSS backup for the example UNIX save point /sp1 (Windows is similar):

- To view the consolidated job log file information following a scheduled backup of /sp1, type the following commands:

- For NetWorker releases previous to NetWorker 9.0.1, type:

```
# tail /nsr/logs/sg/<save group name>/<job#>
```

- For NetWorker 9.0.1, type:

```
# tail /nsr/logs/policy/<policy name>/<workflow name>/<action name>_<sequence#>_logs/<job#>
```

An output similar to the following appears:

```
...
parallel save streams partial completed savetime=1374016342
parallel save streams partial completed savetime=1374016339
parallel save streams partial completed savetime=1374016345
parallel save streams summary test.xyx.com: /sp1 level=full,
311 MB
00:00:08 455 files
parallel save streams summary savetime=1374016345
```

- To list only the master save sets for all /sp1 full and incremental backups, type the following command:

```
# mminfo -ocntR -N "/sp1"-r
"client,name,level,nsavetime,savetime(25),ssid,ssid(53),totalsize,n
files,attrs"
```

- To automatically aggregate <i>/sp1 with /sp1 save sets for browse time-based save point recovery, type the following command:

```
# recover [-t <now or earlier_master_ss_time>] [-d reloc_dir] [-a] /sp1
```

Recommendations to enhance PSS performance

Observe the following recommendations to benefit from PSS performance enhancements.

- The PSS feature boosts backup performance by splitting the save point for PSS into multiple streams that are based on client parallelism. The fairly equal distribution of directory and file sizes in save sets adds additional performance benefit from PSS.
- Large save sets residing on storage with sufficiently high aggregate throughput from concurrent read streams perform significantly better with PSS. Avoid using slow storage with high disk read latency with PSS.
- Ensure that the target devices are fast enough to avoid write contentions or target device queuing since PSS splits a single save point into multiple save streams.
- If the target device is Data Domain, ensure PSS does not saturate the max sessions allowable limit on the DDR. Each Boost device allows a maximum of 60 NetWorker concurrent sessions.
- For NetWorker 9.0.1, keep the client parallelism (CP) value set to the default with PSS-enabled backups. Increasing CP significantly results in sessions timeout due to excessive queuing on the target device and NetWorker server.
- Ensure the total number of available device **max sessions** value is greater than or equal to the total CP values across all concurrently active PSS clients. In NetWorker 9.0.1, for each client whose CP value is not a multiple of four, round up the CP value to the nearest multiple of four when totaling the CP values. Otherwise, some PSS client backups may timeout or be unnecessarily delayed.
- PSS simulates the manual division of a client resource save set entry, or save point, into multiple smaller entries. For example, /sp into /sp, <1>/sp, <2>/sp, and <3>/sp. Once backup completes for save point /sp, its smaller save sets are inherently independent of each other, although they form a logical group. Therefore, manually deleting some but not all of these smaller save sets before expiry leads to subsequent incomplete or failed recovery operations. Avoid manually deleting PSS save sets. If manual deletion is necessary, then always delete the save sets as a complete group.

NetWorker resource considerations

When you create NetWorker workflow and action resources, consider the following recommendations:

- The parallelism value for the action resource should not exceed 25.
- The total number of clients in a single workflow should not exceed 100.

Wrapper script to pass environment variable to save and savegroup commands

In UNIX and Linux environments, if processes initiated by the NetWorker server do not start, use a wrapper script to pass the server's environment variable, for example, *NSR_READ_SIZE* or *NSR_SG_PSS_MAX_SP_SPLIT*, to the *save* and *savegroup* commands.

The wrapper script will look similar to the following:

```
#!/bin/sh
# usage: cd nwbindir; mv nwcmd nwcmd.bin; cp this_script_file nwcmd;
chmod +rx nwcmd
export NW_OR_OS_REQUIRED_ENVVAR=xxx
exec /nwbindir/nwcmd.bin # -D9 -vvvvvv
```

Example 2 Save command wrapper script for NSR_READ_SIZE

The following example demonstrates a save command wrapper script for the NSR_READ_SIZE environment variable.

```
#!/bin/sh
# usage: cd nw_bin_dir; mv save save.bin; cp this_file save; chmod
+rx this_file
export NSR_READ_SIZE=131072 #128KB=131072, 32KB=32768, 64KB=default
exec /nw_bin_path/save.bin # -D9 -vvvvvv
```

Example 3 Savegrp wrapper script for NSR_SG_PSS_MAX_SP_SPLIT

The following example demonstrates a savegroup wrapper script for the NSR_SG_PSS_MAX_SP_SPLIT environment variable.

```
#!/bin/sh
# usage: cd nw_bin_dir; mv savegrp savegrp.bin; cp this_file savegrp;
chmod +rx this_file
export NSR_SG_PSS_MAX_SP_SPLIT=8
exec /nw_bin_path/savegrp.bin # -D9 -vvvvvv
```

Internal maintenance task requirements

Requirements for completing maintenance tasks can add significant load to the NetWorker software:

- Daily index and media database consistency checks adds 40 IOPS for small environments, and up to 200 IOPS for large environments with more than 1,000 configured clients.
- Environments with very long backup and retention times (1 year or more) experience large internal database growth resulting in additional requirements of up to 100 to 200 IOPS.
- Purge operations can take 30 IOPS for small environments with up to 1000 backup jobs per day, 100 IOPS for mid-size environments and up to 200 IOPS for large environments with high loads of 50,000 jobs per day.

Reporting task requirements

Monitoring tools like the NMC server, DPA, custom reporting, or monitoring scripts contribute to additional load on the NetWorker server:

Procedure

- For each NMC server, add an additional 100 IOPS.
- For DPA reporting, add an additional 250 IOPS.

Results

Customer reporting or monitoring scripts can contribute significant load depending on the design. For example, continuous reporting on the NetWorker index and media databases can add up to 500 IOPS.

Manual NetWorker server task requirements

Manual tasks on the NetWorker server can add additional load:

- Each recover session that must enumerate objects on the backup server adds additional load to the NetWorker server.
For example, to fully enumerate 10,000 backup jobs, the NetWorker server can require up to 500 IOPS.
- For spikes, and unrelated operating system workloads, the total number of calculated IOPS should be increased by 30%.
- Single disk performance is often insufficient for large NetWorker servers. The following table provides information on single disk performance.
To achieve higher IOPS, combine multiple disks for parallel access. The best performance for standard disks is achieved with RAID 0+1. However, modern storage arrays are often optimized for RAID 5 access for random workloads on the NetWorker server. Hardware-based write-back cache can significantly improve NetWorker server performance. The following table provides guidelines on the NetWorker server IOPS requirements.

Table 7 Required IOPS for NetWorker server operations

Type of operation	Small NetWorker environment (1)	Medium NetWorker environment (2)	Large NetWorker environment (3)
Concurrent backups	30	80	170
Bootstrap backups	50	150	400
Backup group startup	50	150	250
Volume management	0	0	100
Large NDMP backups	100	100	200
Standard daily maintenance tasks	40	75	100
Large internal database maintenance	0	100	200
Purge operations	50	150	300
NMC reporting	50	75	100
DPA reporting	50	100	250
Recovery	30	200	500

(1) A small NetWorker server environment is considered to have less than 100 clients, or 100 concurrent backup sessions.

(2) A medium NetWorker server environment is considered to have more than 100, and up to 400 clients or 250 concurrent backup sessions.

(3) A large NetWorker server environment is considered to have more than 400 clients, or 500 concurrent backup sessions.

IOPS considerations

There are considerations and recommendations for IOPS values.

- The NetWorker software does not limit the number of clients per datazone, but a maximum of 1000 clients is recommended due to the complexity of managing large datazones, and the increased hardware requirements on the NetWorker server.

NOTICE

As the I/O load on the NetWorker server increases, so does the storage layer service time. If service times exceed the required values there is a direct impact on NetWorker server performance and reliability. Information on the requirements for maximum service times are available in [NetWorker server and storage node disk write latency on page 25](#).

- The NetWorker server performs the data movement itself. If the backup device resides on the server rather than the NetWorker storage node, the backup performance is directly impacted.
Examples 2 and 3 are based on the preceding requirements that are listed in Table 7.

Example 4 Small to medium NetWorker datazone

- Optimized: 200 clients running in parallel with the following characteristics:
 - 100 jobs with up to 1,000 backup jobs per day.
 - Backups spread over time.
 - No external reporting.
 - No overlapping maintenance tasks.
 - Minimum required IOPS: 200, recommended IOPS: 400.
- Non-optimized: the same workload, however:
 - Most backup jobs start simultaneously.
 - Production backups overlap bootstrap and maintenance jobs.
 - Additional reporting is present.
 - Minimum required IOPS: 800, recommended IOPS 1000.

Example 5 Large NetWorker datazone

- Optimized: 1000 clients running in parallel with the following characteristics:
 - 500 jobs with up to 50,000 backup jobs per day.
 - Backups spread over time.
 - Backups using backup to disk, or large tape volumes.
 - No external reporting.
 - No overlapping maintenance tasks.
 - Minimum required IOPS: 800, recommended IOPS: 1000.
- Non-optimized: the same workload, however:
 - Most backup jobs start simultaneously.
 - Many small volumes are used.
 - Production backups overlap bootstrap and maintenance jobs.
 - Additional reporting is present.
 - Minimum required IOPS: 2000, recommended IOPS: 2500.

Note

This example identifies that the difference in NetWorker configuration can result in up to a 250% additional load on the NetWorker server. Also, the impact on sizing is such that well-optimized large environments perform better than non-optimized medium environments.

IOPS values for disk drive technologies

The disk drive type determines the IOPS values for randoms small blocks and sequential large blocks.

The following table lists disk drive types and their corresponding IOPS values.

Table 8 Disk drive IOPS values

Disk drive type	Values per device
Enterprise Flash Drives (EFD)	2500 IO/s for random small block IOs or 100 MB/s sequential large blocks
Fibre Channel drives (FC drives (15k RPM))	180 IO/s for random small block IOs or 12 MB/s sequential large blocks
FC drives (10K RPM)	140 IO/s for random small block IOs or 10 MB/s sequential large blocks
SATA2 or LCFC (7200 RPM)	80 IO/s for random small block IOs or 8 MB/s sequential large blocks
SATA drives (7200 RPM)	60 IO/s for random small block IOs or 7 MB/s sequential large blocks
PATA drives (5400 RPM)	40 IO/s for random small block IOs or 7 MB/s sequential large blocks

File history processing

File history is processed by NDMP at the end of the backup operation, rather than during the backup. This results in perceived long idle times.

The actual file history processing time is linear despite the number of files in the dataset. However, the processing time depends on other storage system factors, such as:

- The RAID type
- The number of disks being configured
- The cache size
- The type of file system for hosting `/nsr/index` and `/nsr/tmp`

NOTICE

The expected results are approximately 20 minutes per each 10 million files.

File history processing creates a significant I/O load on the backup server, and increases IOPS requirements by 100-120 I/O operations per second during processing. If minimum IOPS requirements are not met, file history processing can be significantly slower.

Network

Several components impact network configuration performance:

- **IP network:**
A computer network made of devices that support the Internet Protocol to determine the source and destination of network communication.
- **Storage network:**
The system on which physical storage, such as tape, disk, or file system resides.
- **Network speed:**
The speed at which data travels over the network.
- **Network bandwidth:**
The maximum throughput of a computer network.
- **Network path:**
The communication path used for data transfer in a network.
- **Network concurrent load:**
The point at which data is placed in a network to ultimately maximize bandwidth.
- **Network latency:**
The measure of the time delay for data traveling between source and target devices in a network.

Target device

Storage type and connectivity have the types of components that impact performance in target device configurations.

- **Storage type:**
 - **Raw disk versus Disk Appliance:**
 - Raw disk: Hard disk access at a raw, binary level, beneath the file system level.
 - Disk Appliance: A system of servers, storage nodes, and software.
 - **Physical tape versus virtual tape library (VTL):**
 - VTL presents a storage component (usually hard disk storage) as tape libraries or tape drives for use as storage medium with the NetWorker software.

- Physical tape is a type of removable storage media, generally referred to as a volume or cartridge, that contains magnetic tape as its medium.
- Connectivity:
 - Local, SAN-attached:
A computer network, separate from a LAN or WAN, designed to attach shared storage devices such as disk arrays and tape libraries to servers.
 - IP-attached:
The storage device has its own unique IP address.

The component 70 percent rule

Manufacturer throughput and performance specifications based on theoretical environments are rarely, or never achieved in real backup environments. It is a best practice to never exceed 70 percent of the rated capacity of any component.

Components include:

- CPU
- Disk
- Network
- Internal bus
- Memory
- Fibre Channel

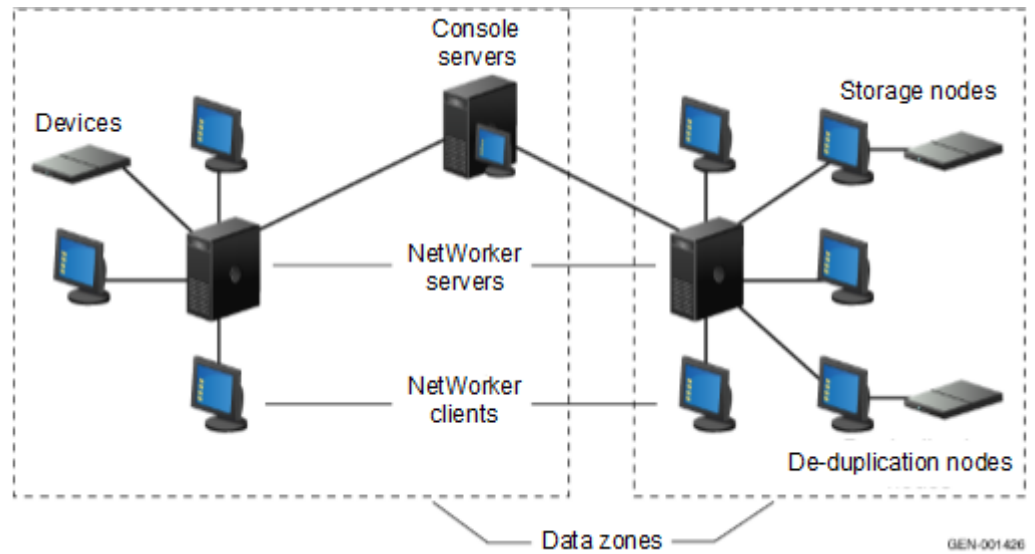
Performance and response time significantly decrease when the 70 percent utilization threshold is exceeded.

The physical tape drives, and solid state disks are the only exception to this rule, and should be used as close to 100 percent as possible. Neither the tape drives, nor the solid state disks suffer performance degradation during heavy use.

Components of a NetWorker environment

A NetWorker datazone is constructed of several components.

The following figure illustrates the main components in a NetWorker environment. The components and technologies that make up a NetWorker environment are listed below.

Figure 4 NetWorker datazone components

Datazone

A datazone is a single NetWorker server and its client computers. Additional datazones can be added as backup requirements increase.

NOTICE

It is recommended to have no more than 1500 clients or 3000 client instances per NetWorker datazone. This number reflects an average NetWorker server and is not a hard limit.

NetWorker Management Console

The NetWorker Management Console (NMC) is used to administer the backup server and it provides backup reporting capabilities.

The NMC often runs on the backup server, and adds significant load to the backup server. For larger environments, it is recommended to install NMC on a separate computer. A single NMC server can be used to administer multiple backup servers.

Components that determine NMC performance

Some components determine the performance of NMC:

- TCP network connectivity to backup server: All communication between NMC and NW server is over TCP and such high-speed low-latency network connectivity is essential.
- Memory: Database tasks in larger environments are memory intensive, make sure that NMC server is equipped with sufficient memory.
- CPU: If the NMC server is used by multiple users, make sure that it has sufficient CPU power to ensure that each user is given enough CPU time slices.

Minimum system requirements for the NMC server

Available disk space and JRE with Web Start must meet specific minimum requirements for the NMC server:

- Memory:

A minimum 1 GHz with 512 MB of RAM is required. Add an additional 512 MB RAM to run reports.

- Available disk space:
Dual core 2 GHz and 2 GB of RAM with a buffer of disk space for a large Console database with multiple users.
- JRE with Web Start:
55 MB, and as the number of NetWorker monitored servers increases, increase the processor capabilities:
 - For 50 servers: Dual 1 GHz with no less than 2 GB RAM
 - For 100 servers: Dual 1 GHz with no less than 4 GB RAM
 - For 200 servers: Quad 1 GHz with no less than 8 GB RAM

Console database

Use formulas to estimate the size and space requirements for the Console database.

Formula for estimating the size of the NetWorker Management Console database

The Console server collects data from the NetWorker servers in the enterprise, and stores the data in its local Console database.

By default, the database is installed on the local file system that can provide the most available space. The Console integrates and processes this information to produce reports that facilitate trend analysis, capacity planning, and problem detection. The *EMC NetWorker Administration Guide* provides information about reports.

To store the collected data, allocate sufficient disk space for the Console database. Several factors affect the amount of disk space required:

- The number of NetWorker servers monitored for the reports
- The number of savegroups run by each of those servers
- The frequency with which savegroups are run
- The length of time report data is saved (data retention policies)

NOTICE

Since the amount of required disk space is directly related to the amount of historical data stored, the requirements can vary greatly, on average between 0.5 GB and several GB. Allow space for this when planning hardware requirements.

Formulas for estimating the space required for the Console database information

There are existing formulas used to estimate the space needed for different types of data and to estimate the total space required.

Save set media database

To estimate the space needed for the save set media database, multiply the weekly amount of save sets by the number of:

- NetWorker servers monitored by the Console
- Weeks in the Save Set Output policy

The result indicates the length of time that a save set took to run successfully. The results also identify the number of files that were backed up, and how much data was saved during the operation.

Save set output

To estimate the space needed for the save set media database, multiply the weekly amount of output messages by the number of:

- NetWorker servers monitored by the Console
- Save Set Output Retention policy

The result indicates how many groups and save sets were attempted and their success or failure.

Savegroup completion data

To estimate the space needed for the save set media database, multiply the weekly amount of savegroups by the number of:

- NetWorker servers monitored by the Console
- Weeks in the Completion Data Retention policy

The result can be used to troubleshoot backup problems.

NetWorker server

NetWorker servers provide services to back up and recover data for the NetWorker client computers in a datazone. The NetWorker server can also act as a storage node and control multiple remote storage nodes.

Index and media management operations are some of the primary processes of the NetWorker server:

- The client file index tracks the files that belong to a save set. There is one client file index for each client.
- The media database tracks:
 - The volume name
 - The location of each save set fragment on the physical media (file number/file record)
 - The backup dates of the save sets on the volume
 - The file systems in each save set
- Unlike the client file indexes, there is only one media database per server.
- The client file indexes and media database can grow to become prohibitively large over time and will negatively impact backup performance.
- The NetWorker server schedules and queues all backup operations, tracks real-time backup and restore related activities, and all NMC communication. This information is stored for a limited amount of time in the jobsdb which for real-time operations has the most critical backup server performance impact.

NOTICE

The data stored in this database is not required for restore operations.

Components that determine backup server performance

The `nsrmmdbd` process uses a CPU intensive operation when thousands of save sets are processed in a single operation. Therefore, cloning operations with large save sets should be run outside of the primary backup window. Also, any NetWorker maintenance activities should be run outside of the primary backup window.

Some components that determine NetWorker server backup performance are:

- Use a 64-bit system for the NetWorker server.
- Use current hardware for the NetWorker server. For example, the current version of the NetWorker server software will not operate well on hardware built more than 10 years ago.
- Minimize these system resource intensive operations on the NetWorker server during heavy loads, such as a high number of concurrent backup/clone/recover streams:
 - `nsrim`
 - `nsrck`
- The disk used to host the NetWorker server (`/nsr`):
The typical NetWorker server workload is from many small I/O operations. This is why disks with high latency perform poorly despite having peak bandwidth. High latency rates are the most common bottleneck of a backup server in larger environments.
- Avoid additional software layers as this adds to storage latency. For example, the antivirus software should be configured with the NetWorker databases (`/nsr`) in its exclusion list.
- Plan the use of replication technology carefully as it significantly increases storage latency.
- Ensure that there is sufficient CPU power for large servers to complete all internal database tasks.
- Use fewer CPUs, as systems with fewer high performance CPUs outperform systems with numerous lower performance CPUs.
- Do not attach a high number of high performance tape drives or AFTD devices directly to a backup server.
- Ensure that there is sufficient memory on the server to complete all internal database tasks.
- Off-load backups to dedicated storage nodes when possible for clients that must act as a storage node by saving data directly to backup server.

NOTICE

The system load that results from storage node processing is significant in large environments. For enterprise environments, the backup server should backup only its internal databases (index and bootstrap).

NetWorker storage node

A NetWorker storage node can be used to improve performance by off loading from the NetWorker server much of the data movement involved in a backup or recovery operation. NetWorker storage nodes require high I/O bandwidth to manage the transfer of data transfer from local clients, or network clients to target devices.

Components that determine storage node performance

Some components determine storage node performance:

- Performance of the target device used to store the backup.
- Connectivity of the system. For example, a storage node used for TCP network backups can save data only as fast as it is able to receive the data from clients.
- I/O bandwidth: Ensure that there is sufficient I/O bandwidth as each storage node uses available system bandwidth. Therefore, the backup performance of all devices is limited by the I/O bandwidth of the system itself.

- CPU: Ensure that there is sufficient CPU to send and receive large amounts of data.
- Do not overlap staging and backup operations with a VTL or AFTD solution by using ATA or SATA drives. Despite the performance of the array, ATA technology has significant performance degradation on parallel read and write streams.

NetWorker client

A NetWorker client computer is any computer whose data must be backed up. The NetWorker Console server, NetWorker servers, and NetWorker storage nodes are also NetWorker clients.

NetWorker clients hold mission critical data and are resource intensive. Applications on NetWorker clients are the primary users of CPU, network, and I/O resources. Only read operations performed on the client do not require additional processing.

Client speed is determined by all active instances of a specific client backup at a point in time.

Components that determine NetWorker client performance

Some components determine NetWorker client performance:

- Client backups are resource intensive operations and impact the performance of primary applications. When sizing systems for applications, be sure to consider backups and the related bandwidth requirements. Also, client applications use a significant amount of CPU and I/O resources slowing down backups. If a NetWorker client does not have sufficient resources, both backup and application performance are negatively impacted.
- NetWorker clients with millions of files. As most backup applications are file based solutions, a lot of time is used to process all of the files created by the file system. This negatively impacts NetWorker client backup performance. For example:
 - A full backup of 5 million 20 KB files takes much longer than a backup of a half million 200 KB files, although both result in a 100 GB save set.
 - For the same overall amount of changed data, an incremental/differential backup of one thousand 100 MB files with 50 modified files takes much less time than one hundred thousand 1 MB files with 50 modified files.
- Encryption and compression are resource intensive operations on the NetWorker client and can significantly affect backup performance.
- Backup data must be transferred to target storage and processed on the backup server:
 - Client/storage node performance:
 - A local storage node: Uses shared memory and does not require additional overhead.
 - A remote storage node: Receive performance is limited by network components.
 - Client/backup server load:
Does not normally slow client backup performance unless the backup server is significantly undersized.

NetWorker databases

Several factors determine the size of NetWorker databases.

These factors are available in [NetWorker database bottlenecks on page 52](#).

Optional NetWorker Application Modules

NetWorker Application Modules are used for specific online backup tasks.

Additional application-side tuning might be required to increase application backup performance. The documentation for the applicable NetWorker module provides details.

Virtual environments

NetWorker clients can be created for virtual machines for either traditional backup or VADP.

Additionally, the NetWorker software can automatically discover virtual environments and changes to those environments on either a scheduled or on-demand basis and provides a graphical view of those environments.

NetWorker deduplication nodes

A NetWorker deduplication node is an EMC Avamar® server that stores deduplicated backup data.

The initial backup to a deduplication node should be a full backup. During subsequent backups, the Avamar infrastructure identifies redundant data segments at the source and backs up only unique segments. This reduces the time required to perform backups, as well as both the network bandwidth and storage space used for backups of the NetWorker Management Console.

Recovery performance factors

Recovery performance can be impeded by network traffic, bottlenecks, large files, and other factors.

Some considerations for recovery performance are:

- File-based recovery performance depends on the performance of the backup server, specifically the client file index. Information on the client file index is available in [NetWorker server on page 44](#).
- The fastest method to recover data efficiently is to run multiple `recover` commands simultaneously by using `save set recover`. For example, three `save set recover` operations provide the maximum possible parallelism given the number of processes, the volume, and the save set layout.
- If multiple, simultaneous recover operations run from the same tape, be sure that the tape does not mount and start until all recover requests are ready. If the tape is used before all requests are ready, the tape is read multiple times slowing recovery performance.
- Multiplexing backups to tape slows recovery performance.

Connectivity and bottlenecks

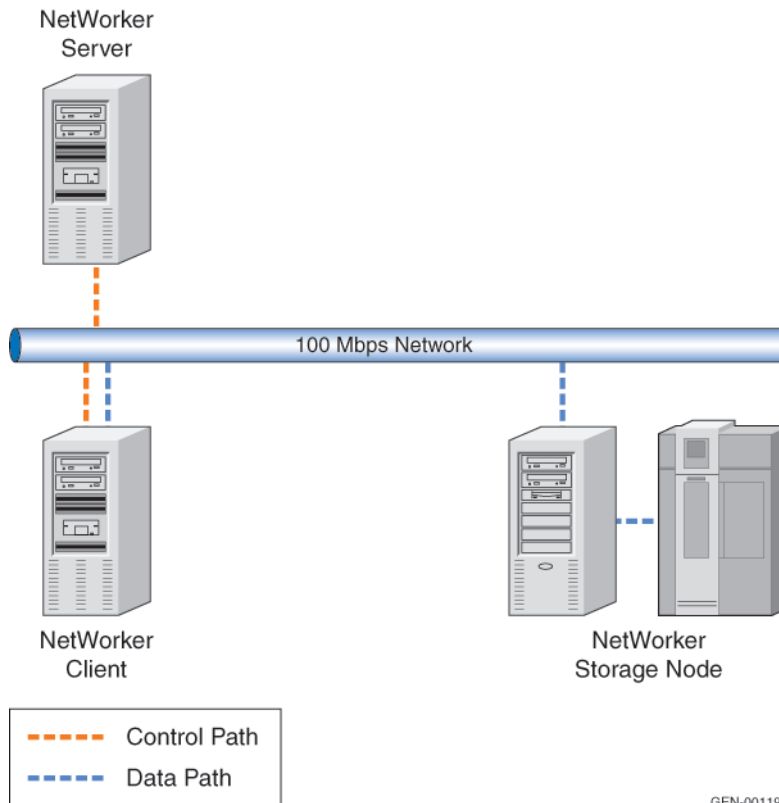
The backup environment consists of various devices from system, storage, network, and target device components, with hundreds of models from various vendors available for each of them.

The factors affecting performance with respect to connectivity are listed here:

- Components can perform well as standalone devices, but how well they perform with the other devices on the chain is what makes the configuration optimal.

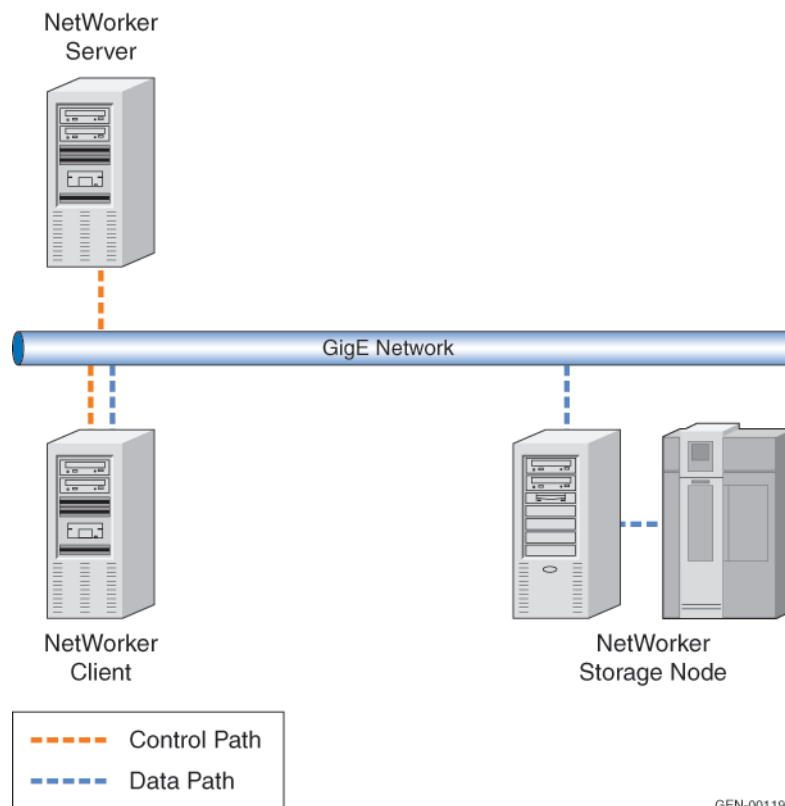
- Components on the chain are of no use if they cannot communicate to each other.
- Backups are data intensive operations and can generate large amounts of data. Data must be transferred at optimal speeds to meet business needs.
- The slowest component in the chain is considered a bottleneck. In the following figure, the network is unable to gather and send as much data as that of the components. Therefore, the network is the bottleneck, slowing down the entire backup process. Any single network device on the chain, such as a hub, switch, or a NIC, can be the bottleneck and slow down the entire operation.

Figure 5 Network device bottleneck



GEN-001194

As illustrated in the following figure, the network is upgraded from a 100 base T network to a GigE network, and the bottleneck has moved to another device. The host is now unable to generate data fast enough to use the available network bandwidth. System bottlenecks can be due to lack of CPU, memory, or other resources.

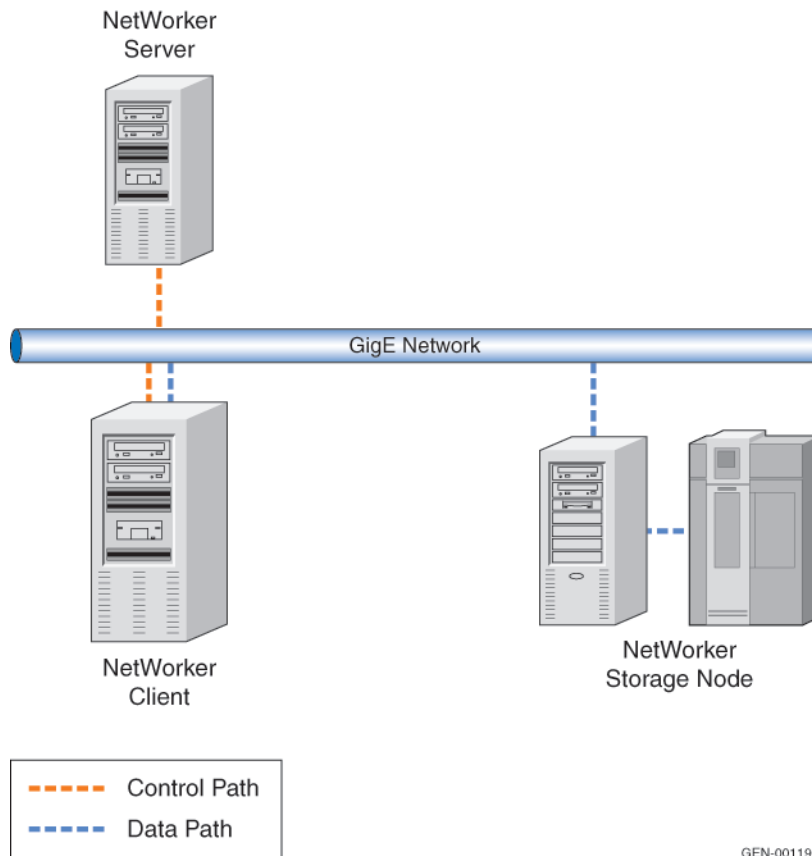
Figure 6 Updated network

GEN-001195

As illustrated in the following figure, the NetWorker client is upgraded to a larger system to remove it as the bottleneck. With a better system and more network bandwidth, the bottleneck is now the target device. Tape devices often do not perform well as other components. Some factors that limit tape device performance are:

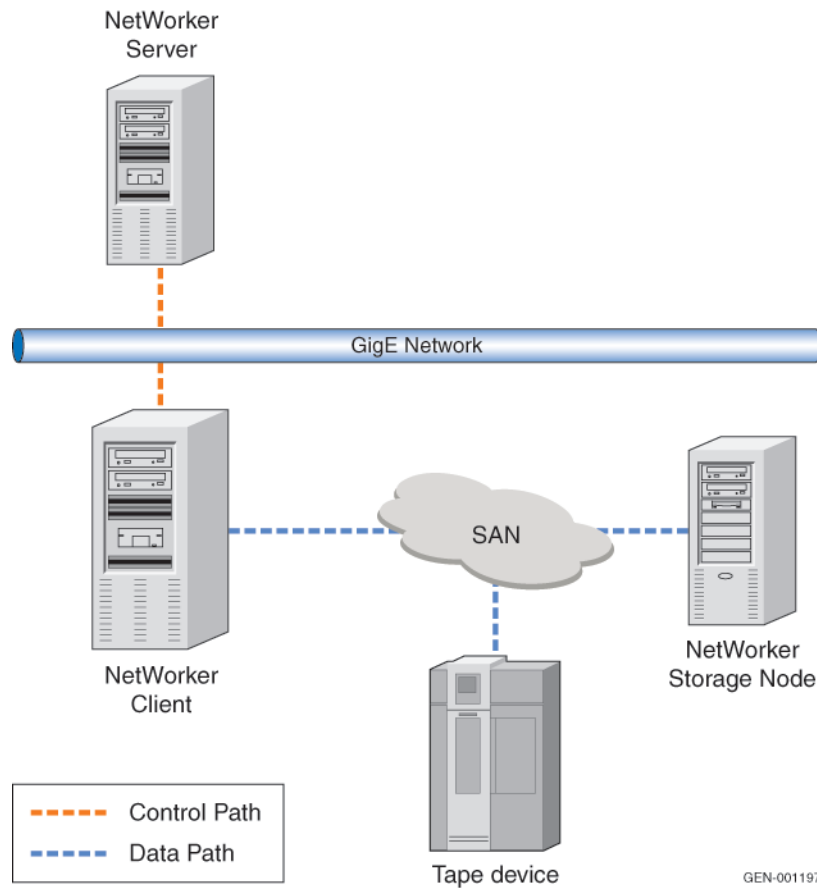
- Limited SCSI bandwidth
- Maximum tape drive performance reached
Improve the target device performance by introducing higher performance tape devices, such as Fibre Channel based drives. Also, SAN environments can greatly improve performance.

Figure 7 Updated client



GEN-001196

As illustrated in the following figure, higher performance tape devices on a SAN remove them as the bottleneck. The bottleneck device is now the storage devices. Although the local volumes are performing at optimal speeds, they are unable to use the available system, network, and target device resources. To improve the storage performance, move the data volumes to high performance external RAID arrays.

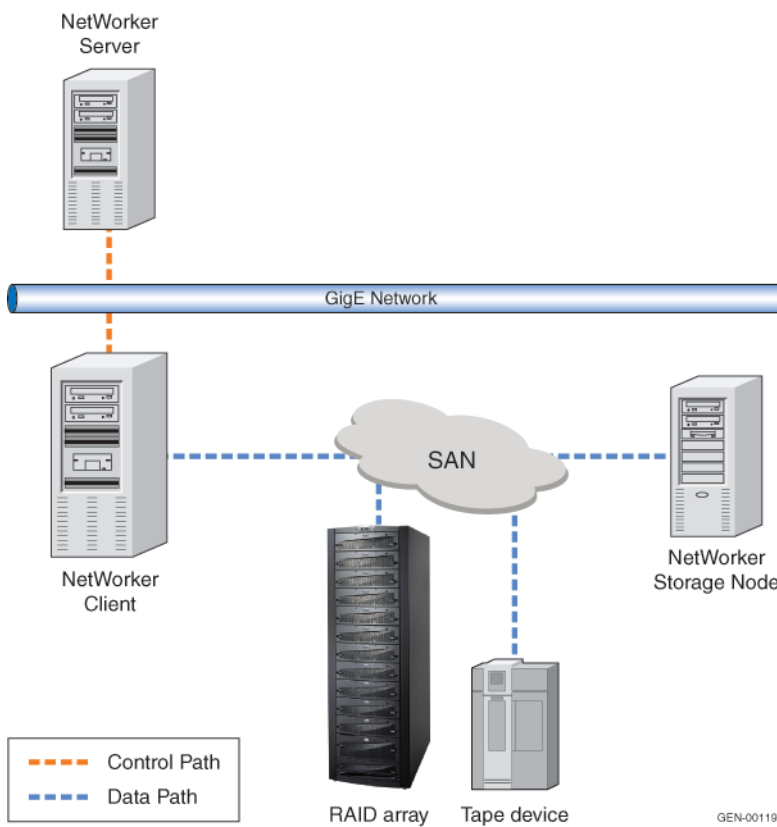
Figure 8 Dedicated SAN

GEN-001197

Although the local volumes are performing at optimal speeds, they are unable to use the available system, network, and target device resources. To improve the storage performance, move the data volumes to high performance external RAID arrays.

As illustrated in the following figure, the external RAID arrays have improved the system performance. The RAID arrays perform nearly as well as the other components in the chain ensuring that performance expectations are met. There will always be a bottleneck, however the impact of the bottleneck device is limited as all devices are performing at almost the same level as the other devices in the chain.

Figure 9 Raid array



NOTICE

This section does not suggest that all components must be upgraded to improve performance, but attempts to explain the concept of bottlenecks, and stresses the importance of having devices that perform at similar speeds as other devices in the chain.

NetWorker database bottlenecks

There are several factors that determine the size of NetWorker databases:

- NetWorker resource database (`nsr/res` or `networker install dir/res`): The number of configured resources.
- NetWorker jobs database (`nsr/res/jobddb`): The number of jobs such as backups, restores, clones multiplied by number of days set for retention. This can exceed 100,000 records in the largest environments and is one of the primary performance bottlenecks. The overall size is never significant.
- For the NetWorker media database (`nsr/mm`): The number of save sets in retention and the number of labeled volumes. In the largest environments this can reach several Gigabytes of data.
- For the NetWorker client file index database (`nsr/index`): The number of files indexed and in the browse policy. This is normally the largest of the NetWorker databases. For storage sizing, use this formula:

$$\text{Index catalog size} = \{[(F+1)*N] + [(I+1) * (DFCR*N)]\} * [(1+G)*C]$$

where:

F = 4 (Full Browse Period set to 4 weeks)

N = 1,000,000 (one million files for this example)

I = 24 (A four week browse period for incremental backups - minus the full backups)

DfCR = 3% (Daily file change rate for standard user file data)

G = 25% (estimated annual growth rate %)

C = 160 bytes (Constant number of bytes per file)

For example:

$$\{[(4+1)*1,000,000] + [(24+1) * (3%*1,000,000)]\} * [(1+.25)*160]$$

$$\{5,000,000 + [25 * 30,000]\} * [1.25 * 160]$$

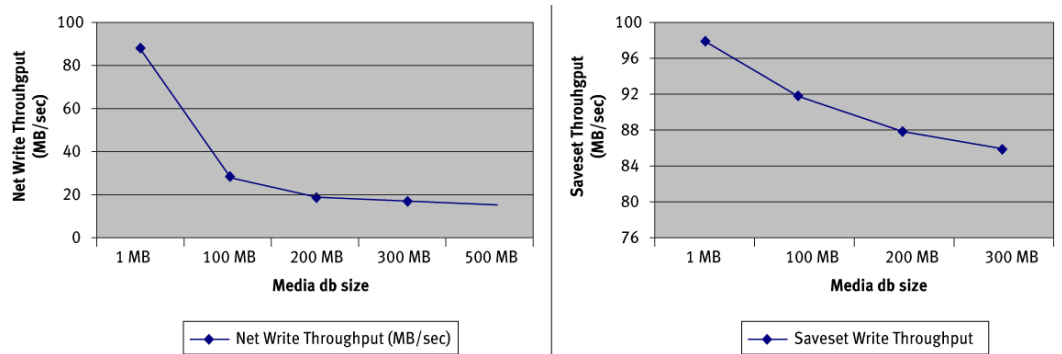
$$5,750,000 * 200 \text{ bytes} = 1,150,000,000 \text{ bytes} = 1150 \text{ MB}$$

NOTICE

The index database can be split over multiple locations, and the location is determined on a per client basis.

The following figure illustrates the overall performance degradation when the disk performance on which NetWorker media database resides is a bottleneck. The chart on the right illustrates net data write throughput (save set + index + bootstrap) and the chart on the left is save set write throughput.

Figure 10 NetWorker server write throughput degradation



CHAPTER 3

Tune Settings

The NetWorker software has various optimization features that can be used to tune the backup environment and to optimize backup and restore performance.

- [Optimize NetWorker parallelism](#) 56
- [File system density](#) 57
- [Disk optimization](#) 58
- [Device performance tuning methods](#) 58
- [Network devices](#) 60
- [Network optimization](#) 64

Optimize NetWorker parallelism

Follow the general best practices for server, group, and client parallelism to ensure optimal performance.

Server parallelism

The server parallelism attribute controls how many save streams the server accepts simultaneously. The more save streams the server can accept, the faster the devices and client disks run. Client disks can run at their performance limit or the limits of the connections between them.

Server parallelism is not used to control the startup of backup jobs, but as a final limit of sessions accepted by a backup server. The server parallelism value should be as high as possible while not overloading the backup server itself.

Client parallelism

Proper client parallelism values are important because backup delays often occur when client parallelism is set too low for the NetWorker server.

The best approach for client parallelism values is:

- For regular clients, use the lowest possible parallelism settings to best balance between the number of save sets and throughput.
- For the backup server, set highest possible client parallelism to ensure that index backups are not delayed. This ensures that groups complete as they should. The best approach to optimize NetWorker client performance is to eliminate client parallelism, reduce it to 1, and increase the parallelism based on client hardware and data configuration.

It is critical that the NetWorker server has sufficient parallelism to ensure index backups do not impede group completion.

The client parallelism values for the client that represents the NetWorker server are:

- Never set parallelism to 1
- For small environments (under 30 servers), set parallelism to at least 8
- For medium environments (31 – 100 servers), set parallelism to at least 12
- For larger environments (100+ servers), set parallelism to at least 16

These recommendations assume that the backup server is a dedicated backup server. The backup server should always be a dedicated server for optimum performance.

Group parallelism

Proper group parallelism values ensure optimal operating system performance.

The best approach for group parallelism values is:

- Create save groups with a maximum of 50 clients with group parallelism enforced. Large save groups with more than 50 clients can result in many operating system processes starting at the same time causing temporary operating system resource exhaustion.
- Stagger save group start times by a small amount to reduce the load on the operating system. For example, it is best to have 4 save groups, each with 50 clients, starting at 5 minute intervals than to have 1 save group with 200 clients.

Note

In NetWorker 9.0, Windows and UNIX clients enabled for parallel save streams (PSS) use the client parallelism value instead of the group parallelism value and, as a result, may exceed the value set for group parallelism. Additionally, Windows non-PSS backups ignore the group parallelism value.

Multiplexing

The Target Sessions attribute sets the target number of simultaneous save streams that write to a device. This value is not a limit, therefore a device might receive more sessions than the Target Sessions attribute specifies. The more sessions specified for Target Sessions, the more save sets that can be multiplexed (or interleaved) onto the same volume.

[AFTD device target and max sessions on page 62](#) provides additional information on device Target Sessions.

Performance tests and evaluation can determine whether multiplexing is appropriate for the system. Follow these guidelines when evaluating the use of multiplexing:

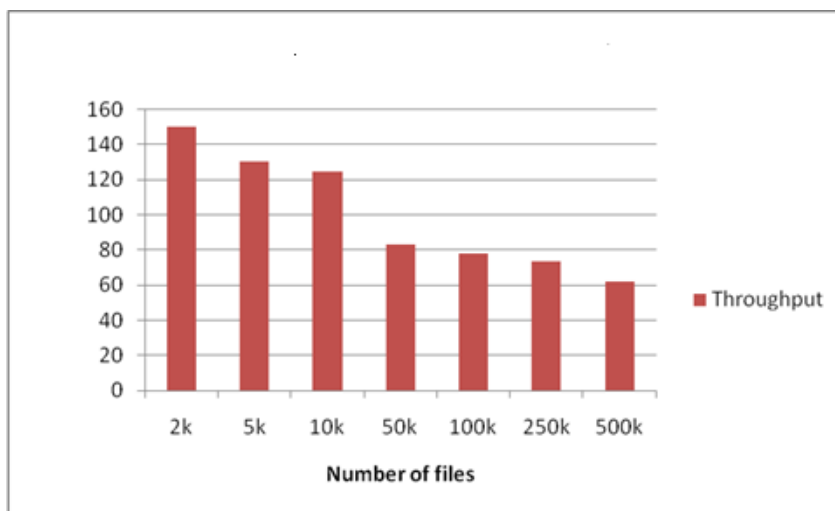
- Find the maximum rate of each device. Use the bigasm test described in [The bigasm directive on page 75](#).
- Find the backup rate of each disk on the client. Use the uasm test described in [The uasm directive on page 76](#).

If the sum of the backup rates from all disks in a backup is greater than the maximum rate of the device, do not increase server parallelism. If more save groups are multiplexed in this case, backup performance will not improve, and recovery performance might slow down.

File system density

File system density has a direct impact on backup throughput.

The NetWorker save operation spends significant time based on file system density specifically when there are many small files. NetWorker performance for high density file systems depends on disk latency, file system type, and number of files in the save set. The following figure illustrates the level of impact file system density has on backup throughput.

Figure 11 Files versus throughput

Disk optimization

NetWorker uses an intelligent algorithm when reading files from a client to choose an optimal block size value in the range of 64 KB and 8 MB based on the current read performance of the client system.

This block size selection occurs during the actual data transfer and does not add any overhead to the backup process, and potentially significantly increases disk read performance.

NOTICE

Read block size is not related to device block size used for backup, which remains unchanged.

This feature is transparent to the rest of the backup process and does not require any additional configuration.

You can override the dynamic block size by setting the `NSR_READ_SIZE` environment variable to a desired value. For example, `NSR_READ_SIZE=65536` forces the NetWorker software to use 64 KB block size during the read process.

Device performance tuning methods

Specific device-related areas can improve performance.

Input/output transfer rate

The I/O rate is the rate at which data is written to a device. Depending on the device and media technology, device transfer rates can range from 500 KB per second to 200 MB per second.

The default block size and buffer size of a device affect its transfer rate. If I/O limitations interfere with the performance of the NetWorker server, try upgrading the device to enable a better transfer rate.

Built-in compression

Turn on device compression to increase effective throughput to the device.

Some 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.

Drive streaming

To obtain peak performance from most 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 it can resume writing. This can cause a delay in the cycle time of a drive, depending on the device.

Device load balancing

Balance data load for simultaneous sessions more evenly across available devices by adjusting target and max sessions per device.

This parameter specifies the minimum number of save sessions to be established before the NetWorker server attempts to assign save sessions to another device. More information on device target and max sessions is available at [AFTD device target and max sessions on page 62](#).

Fragmenting a disk drive

A fragmented file system on Windows clients can cause substantial performance degradation that is based on the amount of fragmentation. Defragment disks to avoid performance degradation.

Procedure

1. Check the file system performance on the client by using a copy or ftp operation without NetWorker to determine if disk fragmentation might be the problem.
2. Run the **Disk Defragmenter** tool on the client to consolidate data so the disk can perform more efficiently:
 - a. Click to open **Disk Defragmenter**.
 - b. Under **Current status**, select the disk to defragment.
 - c. Click **Analyze disk** to verify that fragmentation is a problem. If prompted for an administrator password or confirmation, type the password or provide confirmation.
 - d. When Windows is finished analyzing the disk, check the percentage of fragmentation on the disk in the Last Run column. If the number is above 10%, defragment the disk.
 - e. Click **Defragment disk**. If prompted for an administrator password or confirmation, type the password or provide confirmation.

Note

The defragmentation might take from several minutes to a few hours to complete, depending on the size and degree of fragmentation of the hard disk. You can still use the computer during the defragmentation process.

Network devices

Data that is backed up from remote clients, the routers, network cables, and network interface cards can affect the backup and recovery operations.

This section lists the performance variables in network hardware, and suggests some basic tuning for networks. The following items address specific network issues:

- **Network I/O bandwidth:**
The maximum data transfer rate across a network rarely approaches the specification of the manufacturer because of network protocol overhead.

NOTICE

The following statement concerning overall system sizing must be considered when addressing network bandwidth.

Each attached tape drive (physical VTL or AFTD) uses available I/O bandwidth, and also consumes CPU as data still requires processing.

- **Network path:**
Networking components such as routers, bridges, and hubs consume some overhead bandwidth, which degrades network throughput performance.
- **Network load:**
 - Do not attach a large number of high-speed NICs directly to the NetWorker server, as each IP address uses significant amounts of CPU resources. For example, a mid-size system with four 1 GB NICs uses more than 50 percent of its resources to process TCP data during a backup.
 - Other network traffic limits the bandwidth available to the NetWorker server and degrades backup performance. As the network load reaches a saturation threshold, data packet collisions degrade performance even more.
 - The `nsrmmdbd` process uses a high CPU intensive operation when thousands of save sets are processed in a single operation. Therefore, cloning operations with huge save sets and NetWorker maintenance activities should run outside of the primary backup window.

Fibre Channel latency

To reduce the impact of link latency, increase the NetWorker volume block size.

The result of increased volume block size is that data streams to devices without a frequent need for round-trip acknowledgement.

For low-latency links, increased block size does not have any effect.

For high-latency links, the impact can be significant and will *not* reach the same level of performance as local links.

NOTICE

High bandwidth does not directly increase performance if latency is the cause of slow data.

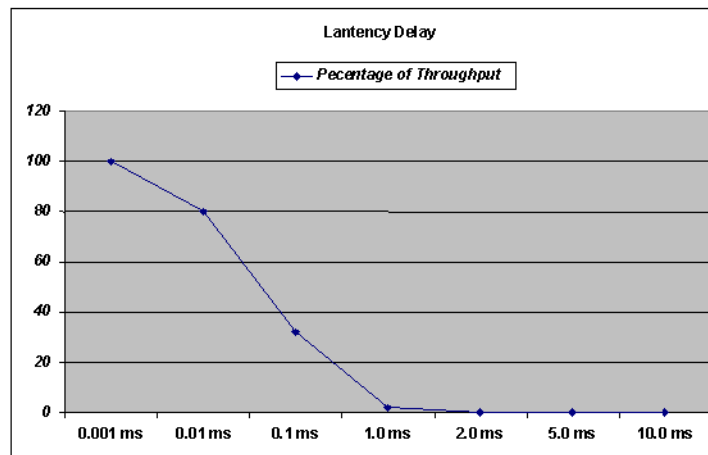
The following table is an example of different block sizes on a physical LTO-4 tape drive connected locally over a 15 KM 8 GB DWDM link.

Table 9 The effect of blocksize on an LTO-4 tape drive

Blocksize	Local backup performance	Remote backup performance
64 KB	173 MB/second	60 MB/second
128 KB	173 MB/second	95 MB/second
256 KB	173 MB/second	125 MB/second
512 KB	173 MB/second	130 MB/second
1024 KB	173 MB/second	130 MB/second

The following figure illustrates that the NetWorker backup throughput drops from 100 percent to 0 percent when the delay is set from 0.001 ms to 2.0 ms.

Figure 12 Fibre Channel latency impact on data throughput



Data Domain

Backup to Data Domain storage can be configured by using multiple technologies:

- NetWorker 8.1 and later supports DD Boost over Fibre Channel. This feature leverages the advantage of the boost protocol in a SAN infrastructure. It provides the following benefits:
 - DD Boost over Fibre Channel (DFC) backup with Client Direct is 20–25% faster when compared to backup with DD VTL.
 - The next subsequent full backup is three times faster than the first full backup.
 - Recovery over DFC is 2.5 times faster than recovery using DD VTL.
- Backup to VTL:
 - NetWorker devices are configured as tape devices and data transfer occurs over Fibre Channel.
 - Information on VTL optimization is available in [Number of virtual device drives versus physical device drives on page 63](#).

- Backup to AFTD over CIFS or NFS:
 - Overall network throughput depends on the CIFS and NFS performance which depends on network configuration. [Network optimization on page 64](#) provides best practices on backup to AFTD over CIFS or NFS.
 - Inefficiencies in the underlying transport limits backup performance to 70-80% of the link speed. For best performance, NetWorker release 7.5 Service Pack 2 or later is required.
- The Client Direct attribute to enable direct file access (DFA) introduced in NetWorker 8.0:
 - Client Direct to Data Domain (DD) using Boost provides much better performance than DFA-AFTD using CIFS/NFS.
 - Backup performance with client direct enabled (DFA-DD/DFA-AFTD) is 20–60% faster than traditional backup using `mmcl`.
 - With an increasing number of streams to single device, DFA handles the backup streams much better than `mmcl`.
- Backup to Data Domain by using a native device type:
 - NetWorker 7.6 Service Pack 1 provides a new device type that is designed specifically for native communication to Data Domain storage over TCP/IP links.
 - With proper network optimization, this protocol can use up to 95 percent of the link speed even at 10 GB/s rates and is the most efficient network transport.
 - In NetWorker 7.6.1, each Data Domain device that is configured in NetWorker is limited to a maximum of ten parallel backup streams. If higher parallelism is required, configure more devices to a limit defined by the NetWorker server edition.
 - In NetWorker 7.6.2 and later, limit number of sessions per device to 60. Despite the method that is used for backup to Data Domain storage, the aggregate backup performance is limited by the maximum ingress rate of the specific Data Domain model.
- The minimum required memory for a NetWorker DataDomain-OST device with each device total streams set to 10 is approximately 160 MB. Each OST stream for BOOST takes an additional 16 MB of memory.
- DD Boost takes between 2% and 40% additional CPU time during backup operations as compared to non-client deduplicated backups for a much shorter period. However, the overall CPU load of a backup to DD Boost is less when compared to traditional `mmd` based backups using CIFS/NFS.

AFTD device target and max sessions

Each supported operating system has a specific optimal Advanced File Type Device (AFTD) device target, and max sessions settings for the NetWorker software.

NetWorker 7.6 and earlier software

The current NetWorker 7.6 and earlier default settings for AFTD target sessions (4) and max sessions (512) are not optimal for AFTD performance.

To optimize AFTD performance for NetWorker 7.6 and earlier, change the default values:

- Set device target sessions from 1.
- Set device max sessions from 32 to avoid disk thrashing.

NetWorker 7.6 Service Pack 1 and later

The defaults for AFTD target sessions and max device sessions are now set to the optimal values for AFTD performance:

- Device target sessions is 1.
- Device max sessions is 32 to avoid disk thrashing.
If required, both Device target, and max session attributes can be modified to reflect values appropriate for the environment.

NetWorker 8.0 and later software

The dynamic `nsrmmmd` attribute in the NSR storage node attribute is off by default for the dynamic provisioning of `nsrmmmd` processes. Turning on the dynamic `nsrmmmd` attribute enables dynamic `nsrmmmd` provisioning.

NOTICE

The dynamic `nsrmmmd` feature for AFTD and DD Boost devices in NetWorker 8.1 is enabled by default. In previous NetWorker versions, this attribute was disabled by default.

When the dynamic `nsrmmmd` attribute is enabled and the number of sessions to a device exceeds the number of target sessions, the visible change in behavior is multiple `nsrmmmd` processes on the same device. This continues until the max `nsrmmmd` count, or max sessions values are reached, whichever is lower.

To turn on backup to disk, select the **Configuration** tab to set these attributes as required:

- **Target Sessions** is the number of sessions the device handles before for another available device is used. For best performance, this should be set to a low value. The default values are four (FTD/AFTD) and six (DD Boost devices) and it may not be set to a value greater than 60.
- **Max Sessions** has a default value of 32 (FTD/AFTD) and 60 (DD Boost devices), which usually provides best performance. It cannot be set to a value greater than 60.
- **Max `nsrmmmd` count** is an advanced setting that can be used to increase data throughput by restricting the number of backup processes that the storage node can simultaneously run. When the target or max sessions are changed, the max `nsrmmmd` count is automatically adjusted according to the formula MS/TS plus four. The default values are 12 (FTD/AFTD) and 14 (DD Boost devices).

NOTICE

It is not recommended to modify both session attributes and max `nsrmmmd` count simultaneously. If you must modify all these values, adjust the sessions attributes first, apply the changes, and then update max `nsrmmmd` count.

Number of virtual device drives versus physical device drives

The acceptable number of virtual devices that are stored on an LTO depends on the type of LTO and the number of planned physical devices.

The following is based on the 70 percent utilization of a Fibre Channel port:

- For LTO-3: three virtual devices for every two physical devices planned.
- For LTO-4: three virtual devices for each physical device planned.
The performance of each of these tape drives on the same port degrades with the number of attached devices.

For example:

- If the first virtual drive reaches the 150 MB per second limit.
- The second virtual drive will not exceed 100 MB per second.
- The third virtual drive will not exceed 70 MB per second.

Network optimization

Adjust the following components of the network to ensure optimal performance.

Advanced configuration optimization

The default TCP operating system parameters are tuned for maximum compatibility with legacy network infrastructures, but not for maximum performance. Thus, some configuration is necessary.

Chapter 4, "Communication Security Settings", in the *EMC NetWorker Security Configuration Guide* provides instructions on advanced configuration options such as multi-homed systems, trunking, and so on.

Operating system TCP stack optimization

There are general and environmental capability rules to ensure operating system TCP stack optimization.

The common rules for optimizing the operating system TCP stack for all use cases are listed here:

- Disable software flow control.
- Increase TCP buffer sizes.
- Increase TCP queue depth.
- Use PCIeExpress for 10 GB NICs. Other I/O architectures do not have enough bandwidth.
More information on PCIeExpress is available in [PCI-X and PCIeExpress considerations on page 22](#).

Rules that depend on environmental capabilities are listed here:

- Some operating systems have internal auto-tuning of the TCP stack. This produces good results in a non-heterogeneous environment. However, for heterogeneous, or routed environments disable TCP auto-tuning.
- Enable jumbo frames when possible. Information on jumbo frames is available in [Jumbo frames on page 67](#).

NOTICE

It is required that *all* network components in the data path are able to handle jumbo frames. Do not enable jumbo frames if this is not the case.

- TCP hardware offloading is beneficial if it works properly. However it can cause CRC mis-matches. Be sure to monitor for errors if it is enabled.
- TCP windows scaling is beneficial if it is supported by all network equipment in the chain.
- TCP congestion notification can cause problems in heterogeneous environments. Only enable it in single operating system environments.

Advanced tuning

IRQ processing for high-speed NICs is very expensive, but can provide enhanced performance by selecting specific CPU cores. Specific recommendations depend on the CPU architecture.

Expected NIC throughput values

High speed NICs are significantly more efficient than common NICs.

Common NIC throughput values are in the following ranges:

- 100 MB link: 6–8 MB/s
- 1 GB link: 45–65 MB/s
- 10 GB link: 150–350 MB/s

With optimized values, throughput for high-speed links can be increased to the following:

- 100 MB link: 12 MB/s
- 1 GB link: 110 MB/s
- 10 GB link: 1100 MB/s

The theoretical maximum throughput for a 10 GB Ethernet link is 1.164 GB/s per direction calculated by converting bits to bytes and removing the minimum Ethernet, IP and TCP overheads.

Network latency

Increased network TCP latency has a negative impact on overall throughput, despite the amount of available link bandwidth. Longer distances or more hops between network hosts can result in lower overall throughput.

Network latency has a high impact on the efficiency of bandwidth use.

For example, The following figures illustrate backup throughput on the same network link, with varying latency.

For these examples, non-optimized TCP settings were used.

Figure 13 Network latency on 10/100 MB per second

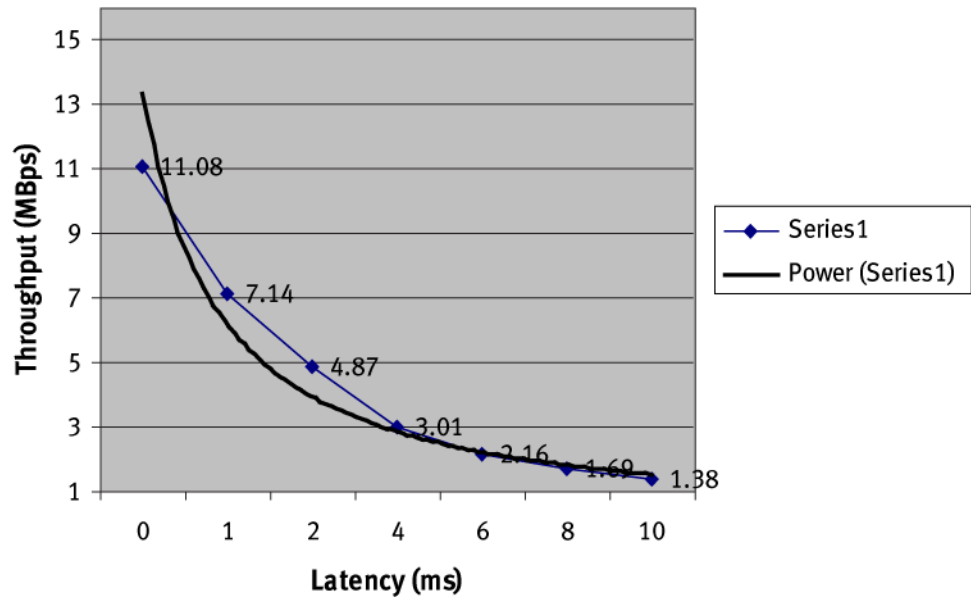
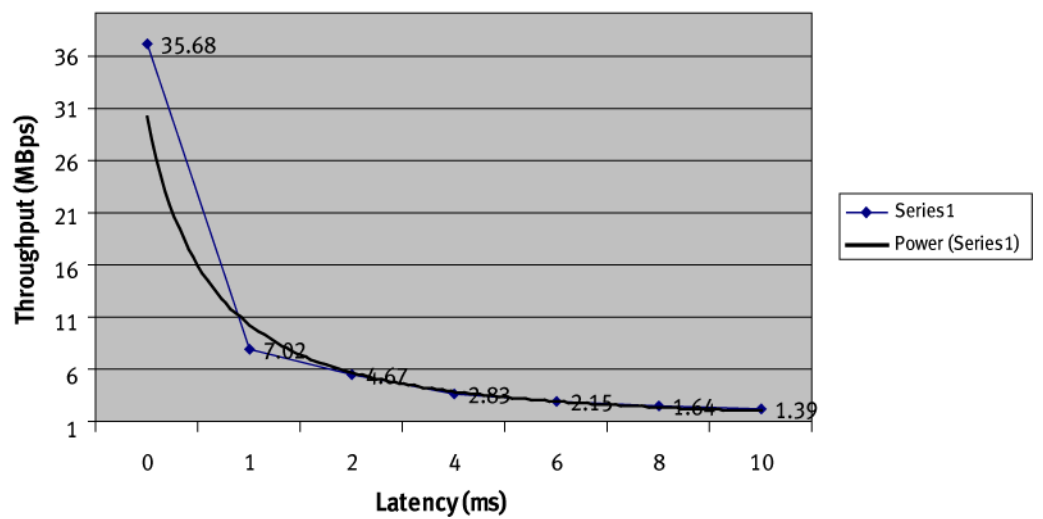


Figure 14 Network latency on 1 GB



Ethernet duplexing

Network links that perform in half-duplex mode cause decreased NetWorker traffic flow performance.

For example, a 100 MB half-duplex link results in backup performance of less than 1 MB per second.

The default configuration setting on most operating systems for duplexing is automatically negotiated as recommended by IEEE802.3. However, automatic negotiation requires that the following conditions are met:

- Proper cabling
- Compatible NIC adapter
- Compatible switch

Automatic negotiation can result in a link performing as half-duplex.

To avoid issues with auto negotiation, force full-duplex settings on the NIC. Forced full-duplex setting must be applied to both sides of the link. Forced full-duplex on only one side of the link results in failed automatic negotiation on the other side of the link.

Firewalls

The additional layer on the I/O path in a hardware firewall increases network latency, and reduces the overall bandwidth use.

Use software firewalls on the backup server as it processes many packets that result in significant overhead.

Details on firewall configuration and impact are available in the *EMC NetWorker Administration Guide*.

Jumbo frames

Use jumbo frames in environments capable of handling them. If both the source, the computers, and all equipment in the data path can handle jumbo frames, increase the MTU to 9 KB.

These examples are for Linux and Solaris operating systems:

- On Linux, type the following command to configure jumbo frames:

```
ifconfig eth0 mtu 9000 up
```

- On Solaris, type the following command to configure jumbo frames for a `nxge` device:

```
ndd -set /dev/nxge<#> accept-jumbo 1
```

where `<#>` is replaced with the driver instance number.

To determine the instance number of a following device, type the following command:

```
nxge /etc /path_to_inst
```

Congestion notification

Methods to disable congestion notification algorithms vary based on the operating system.

On Windows Server 2008, 2008 R2, 2012, and 2012 R2:

- Disable optional congestion notification algorithms by typing the following command:

```
C:\> netsh interface tcp set global ecncapability=disabled
```

- Compound TCP is an advanced TCP algorithm that provides the best results on Windows via the TCP Global parameter *Add-On Congestion Control Provider*. The value for this parameter is `none` if Compound TCP is disabled, or `ctcp` if Compound TCP is enabled.

If both sides of the network conversion are not capable of the negotiation, you can disable *Add-On Congestion Control Provider* by typing the following command:

```
C:\> netsh interface tcp set global congestionprovider=none
```

Note

A reboot of the system is required if you enable *Add-On Congestion Control Provider* by typing the command `C:\> netsh int tcp set global congestionprovider=ctcp`.

On Linux systems:

- Check for non-standard algorithms by typing the following command:

```
cat /proc/sys/net/ipv4/tcp_available_congestion_control
```

- Type the following command to disable ECN:

```
echo 0 >/proc/sys/net/ipv4/tcp_ecn
```

On Solaris systems:

- Type the following command to disable TCP Fusion, if present:

```
set ip:do_tcp_fusion = 0x0
```

TCP buffers

When the rate of inbound TCP packets is higher than the system can process, the operating system drops some of the packets. This scenario can lead to an undetermined NetWorker state and unreliable backup operations. For NetWorker server or storage node systems that are equipped with high-speed interfaces, it is critical to monitor the system TCP statistics for dropped TCP packets, commonly done by using the `netstat -s` command. To avoid dropped TCP packets, increase the TCP buffer size. Depending on the operating system, this parameter is referred to as buffer size, queue size, hash size, backlog, or connection depth.

For high-speed network interfaces, increase size of TCP send/receive buffers.

- Linux:

To modify the TCP buffer settings on Linux:

1. Add the following parameters to the `/etc/sysctl.conf` file:

```
net.core.rmem_default = 262144
net.core.wmem_default = 262144
net.core.rmem_max = 16777216
net.core.wmem_max = 16777216
net.ipv4.tcp_rmem = 8192 524288 16777216
net.ipv4.tcp_wmem = 8192 524288 16777216
```

2. Type the following command:

```
/sbin/sysctl -p
```

3. Set the recommended RPC value:

```
sunrpc.tcp_slot_table_entries = 64
```

4. Enable dynamic TCP window scaling which requires compatible equipment in the data path:

```
sysctl -w net.ipv4.tcp_window_scaling=1
```

- Solaris:

To modify the TCP buffer settings on Solaris, type the following command:

```
tcp_max_buf 10485760
tcp_cwnd_max 10485760
```

```
tcp_recv_hiwat 65536
tcp_xmit_hiwat 65536
```

- AIX:

To modify the TCP buffer settings on AIX, modify the values for the parameters in `/etc/rc.net` if the values are lower than the recommended. Consider the following:

 - The number of bytes a system can buffer in the kernel on the receiving sockets queue:


```
no -o tcp_recvspace=524288
```
 - The number of bytes an application can buffer in the kernel before the application is blocked on a send call:


```
no -o tcp_sendspace=524288
```
- Windows:

The default Windows buffer sizes are sufficient. To modify the TCP buffer settings on Windows:

 - Set the registry entry:


```
AdditionalCriticalWorkerThreads: DWORD=10
```
 - If the NIC drivers can create multiple buffers or queues at the driver-level, enable it at the driver level. For example, Intel 10 GB NIC drivers by default have RSS Queues set to two, and the recommended value for best performance is 16.
 - Windows Server 2008, 2008 R2, 2012 and 2012 R2 platforms provide a method to auto tune the TCP stack. If a server on the LAN or a network device in the datazone such as a router or switch does not support TCP Windows scaling, backups can fail.

On Windows Server 2008, to avoid failed backups and ensure best NetWorker operations, apply the Microsoft Hotfix KB958015, and set the auto tuning level value to highly restricted:

1. Check the current TCP settings:

```
C:\> netsh interface tcp show global
```

2. If required, restrict the Windows TCP receive side scaling auto tuning level:

```
C:\> netsh interface tcp set global
autotuninglevel=highlyrestricted
```

NOTICE

If the hotfix KB958015 is not applied on Windows Server 2008, set the auto tuning level to disabled, not highly restricted.

Increase TCP backlog buffer size

To increase TCP backlog buffer size, set the connection backlog queue to the maximum value allowed:

```
net.ipv4.tcp_max_syn_backlog = 8192
net.core.netdev_max_backlog = 8192
```

The `net.core.somaxconn` value default is 128. Raise the value substantially to support bursts of requests. For example, to support a burst of 1024 requests, set `net.core.somaxconn` to 1024:

```
net.core.somaxconn = 1024
```

NetWorker socket buffer size

It is important to determine the NetWorker socket buffer size for optimization.

To force the use of a larger TCP send/receive window from NetWorker 7.6.x and earlier, include these settings in the NetWorker start script:

```
NSR SOCK BUF SIZE=65536
export NSR SOCK BUF SIZE
```

- The best TCP socket buffer for a 1 GB network is 64 KB.
- The best TCP socket buffer for a 10 GB network is 256KB. Include this setting in the NetWorker start script:

```
NSR SOCK BUF SIZE=262144
```

NOTICE

It is not required to set the NetWorker socket buffer size in NetWorker 8.0 and later. It is implemented in the software and is set to use a 256 KB socket buffer size.

IRQ balancing and CPU affinity

A high-speed network interface that uses either multiple 1 GB interfaces or one 10 GB interface benefits from disabled IRQ balancing and binding to specific CPU core processing.

NOTICE

The general rule is that only one core per physical CPU should handle NIC interrupts. Use multiple cores per CPU only if there are more NICs than CPUs. Handle transmitting and receiving with the same CPU without exception.

These examples are for Linux and Solaris operating systems:

- Linux:

1. Disable IRQ balancing and set CPU affinity manually:

```
service irqbalance stop
chkconfig irqbalance off
```

2. Tune the CPU affinity for the eth0 interface:

```
grep eth0 /proc/interrupts
```

3. Tune the affinity for the highest to the lowest. For example:

```
echo 80 > /proc/irq/177/smp_affinity
echo 40 > /proc/irq/166/smp_affinity
```

SMP affinity works only for IO-APIC enabled device drivers. Check for the IO-APIC capability of a device by using `cat /proc/interrupts`, or by referencing the device documentation.

- Solaris:

Interrupt only one core per CPU. For example, for a system with 4 CPUs and four cores per CPU, use this command:

```
psradm -i 1-3 5-7 9-11 13-15
```

Additional tuning depends on the system architecture.

Successful settings on a Solaris system with a T1/T2 CPU include the following:

```
ddi_msix_alloc_limit 8
tcp_squeue_wput 1
ip_soft_rings_cnt 64
ip_squeue_fanout 1
```

Some NIC drivers artificially limit interrupt rates to reduce peak CPU use which limits the maximum achievable throughput. If a NIC driver is set for “Interrupt moderation,” disable it for optimal network throughput.

Interrupt moderation

On Windows, for a 10 GB network, it is recommended to disable interrupt moderation for the network adapter to improve network performance.

TCP chimney offloading

For systems with NICs capable of handling TCP packets at a lower level, enable TCP chimney offloading on the operating system to increase overall bandwidth utilization, and decrease the CPU load on the system.

NOTICE

Not all NICs that market offloading capabilities are fully compliant with the standard.

- For Windows Server 2008, 2008 R2, 2012 and 2012 R2, use the following command to enable TCP chimney offloading:

```
C:\> netsh interface tcp set global chimney=enabled
```

- For Windows Server 2008 and 2008 R2, use these commands with additional properties to enable TCP offloading:

```
C:\> netsh interface tcp set global dca=enabled
C:\> netsh interface tcp set global netdma=enabled
```

- For Windows Server 2012 and 2012 R2, use the following command with additional properties to enable TCP offloading:

```
C:\> netsh interface tcp set global dca=enabled
```

Note

Windows Server 2012 and 2012 R2 do not support NetDMA.

- Disable TCP offloading for older generation NIC cards that exhibit problems such as backup sessions that stop responding, failures due to RPC errors, or connection reset (CRC) errors similar to the following:

```
Connection reset by peer
```

Note

TCP chimney offloading can cause CRC mis-matches. Ensure that you consistently monitor for errors when you enable TCP chimney offloading.

Name resolution

The NetWorker server relies heavily on the name resolution capabilities of the operating system.

For a DNS server, set low-latency access to the DNS server to avoid performance issues by configuring, either of the following:

- Local DNS cache
or
- Local non-authoritative DNS server with zone transfers from the main DNS server

Ensure that the server name and the hostnames that are assigned to each IP address on the system are defined in the hosts file to avoid DNS lookups for local hostname checks.

CHAPTER 4

Test Performance

This chapter describes how to test and understand bottlenecks by using available tools including NetWorker programs such as `bigasm` and `uasm`.

- [Determine symptoms](#)..... 74
- [Monitor performance](#).....74
- [Determining bottlenecks by using a generic FTP test](#).....75
- [Testing setup performance using the dd test](#)..... 75
- [Test disk performance by using bigasm and uasm](#).....75
- [TCP window size and network latency considerations](#).....76

Determine symptoms

There are many considerations for determining the reason for poor backup performance.

Ask the following questions to determine the cause for poor performance:

- Is the performance consistent for the entire duration of the backup?
- Do the backups perform better when started at a different time?
- Is it consistent across all save sets for the clients?
- Is it consistent across all clients with similar system configuration using a specific storage node?
- Is it consistent across all clients with similar system configuration in the same subnet?
- Is it consistent across all clients with similar system configuration and applications?

Observe how the client performs with different parameters. Inconsistent backup speed can indicate problems with software or firmware.

For each NetWorker client, answer these questions:

- Is the performance consistent for the entire duration of the backup?
- Is there a change in performance if the backup is started at a different time?
- Is it consistent across all clients using specific storage node?
- Is it consistent across all save sets for the client?
- Is it consistent across all clients in the same subnet?
- Is it consistent across all clients with similar operating systems, service packs, applications?
- Does the backup performance improve during the save or does it decrease?

These and similar questions can help to identify the specific performance issues.

Monitor performance

You can monitor the I/O, disk, CPU, and network performance by using native performance monitoring tools.

The monitoring tools available for performance monitoring include the following:

- **Windows:** `perfmon` program
- **UNIX:** `iostat`, `vmstat`, or `netstat` commands

Unusual activity before, during, and after backups can determine that devices are using excessive resources. By using these tools to observe performance over a period, resources that are consumed by each application, including NetWorker, are clearly identified. If it is discovered that slow backups are due to excessive network use by other applications, this can be corrected by changing backup schedules.

High CPU use is often the result of waiting for external I/O, not insufficient CPU power. This is indicated by high CPU use inside SYSTEM versus user space.

On Windows, if much time is spent on Deferred Procedure Calls, it often indicates a problem with device drivers.

Determining bottlenecks by using a generic FTP test

Without using NetWorker components, you can determine whether the bottleneck is in the network or the tape device by using a generic FTP test.

Procedure

1. Create a large datafile on the NetWorker client and send it to the storage node by using FTP.
2. Make note of the time it takes for the file to transfer.
3. Compare the time noted in step two with current backup performance:
 - a. If the FTP performs much faster than the backups, then the bottleneck might be with the tape devices.
 - b. If the FTP performs at a similar rate, then the bottleneck might be in the network.
4. Compare results by using active FTP versus passive FTP transfer. NetWorker backup performance is greatly impacted by the capabilities of the underlying network and the network packets that are used by the NetWorker software.

Results

If there is large difference in the transfer rate, or one type of FTP transfer has spikes, it might indicate the presence of network components that perform TCP packet re-assembly. This causes the link to perform in half-duplex mode, despite all physical parts that are in full-duplex mode.

Testing setup performance using the dd test

Without using NetWorker components, you can use the generic `dd` test to compare device throughput to the manufacturer's suggested throughput.

Procedure

1. Create a large data file on the storage node and use `dd` to send it to the target device. Type the following command:


```
date; dd if=/tmp/5GBfile of=/dev/rmt/0cbn bs= 1MB; date
```
2. Make note of the time it takes for the file to transfer, and compare it with the current tape performance.

Test disk performance by using bigasm and uasm

The `bigasm` and `uasm` directives are NetWorker based tests used to verify performance.

The bigasm directive

The `bigasm` directive generates a specific sized file, and transfers the file over a network or a SCSI connection. The file is then written to a tape or another target device.

The `bigasm` directive creates a stream of bytes in memory and saves them to the target device that eliminates disk access. This helps to test the speed of NetWorker clients, network, and the tape devices ignoring disk access.

Create a `bigasm` directive to generate a very large save set.

The `bigasm` directive ignores disk access to test the performance of client, network, and tape.

The `uasm` directive

The `uasm` directive reads from the disk at maximum speeds to identify disk based bottlenecks.

For example:

```
uasm -s filename > NUL
```

The `uasm` directive tests disk read speeds, and by writing data to a null device can identify disk-based bottlenecks.

TCP window size and network latency considerations

Increased network TCP latency has a negative impact on overall throughput despite the amount of available link bandwidth.

Longer distances or more hops between network hosts can result in lower overall throughput. Since the propagation delay of the TCP packet depends on the distance between the two locations, increased bandwidth will not help if high latency exists between the two sites.

Throughput also depends on the TCP window size and the amount of latency between links. A high TCP window size generally results in better performance, however, with a high latency link, increasing the TCP window may significantly impact the backup window due to packet loss. Every unsuccessful packet that is sent must be kept in memory and must be re-transmitted in case of packet loss. Therefore, for TCP windows with a high latency link, EMC recommends maintaining the default TCP window size.

The network latency impact on NetWorker backup, clone, and recovery performance depends on the control path and data path:

- Latency between clients and NetWorker server (control path) — The latency impact on the NetWorker control path (metadata update) can vary based on the type of data you process during NetWorker backup and recovery operations. For example, if NetWorker clients and the server are separated by a high latency link, and clients back up a high density file system dataset, the large amount of metadata (file indexes) being sent over the wire impacts the index commit.
- Latency between client and target device (DD) (data path) — Latency between the NetWorker client and the target device significantly impacts throughput. Any packet loss will further impact throughput. The high latency link in the data path affects throughput irrespective of the type of data being processed.

The following section provides best practices and recommendations when using high latency networks such as WAN for NetWorker application data and control paths for backup, clone, and recovery operations.

These examples show the results from using high density datasets (many files but with a low overall size) and large density datasets (a small number of files but with a large overall size) during backup, clone, and restore workflows.

The data layout is as follows:

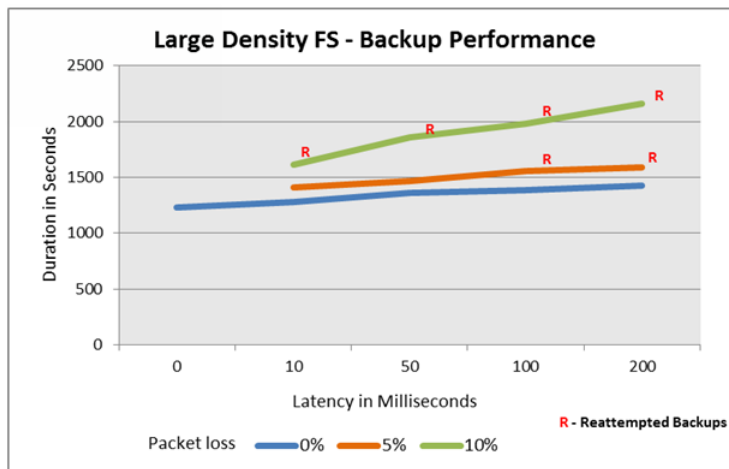
- High density file system (FS): 1 million files with approximately 4000 MB overall size
- Large density file system : <1000 files with approximately 390 GB overall size

Note

These tests were conducted using the WANem simulation tool by inducing latency and packet loss between the NetWorker control and data path. Allow for a 10–20% error margin in the results due to the simulation technique.

WAN Latency impact in control path (NetWorker server and clients separated by high latency) for backup

Figure 15 Large Density FS Backup - WAN between NetWorker Clients and Server (High Packet Loss)



Note

The items that are marked in RED are reattempted backups with client retries >0 and <=5

Figure 16 Large Density FS Backup - WAN between NetWorker Clients and Server (Low Packet Loss)

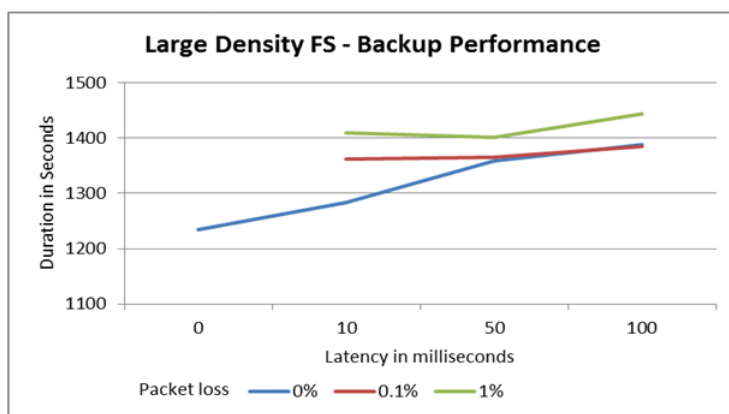
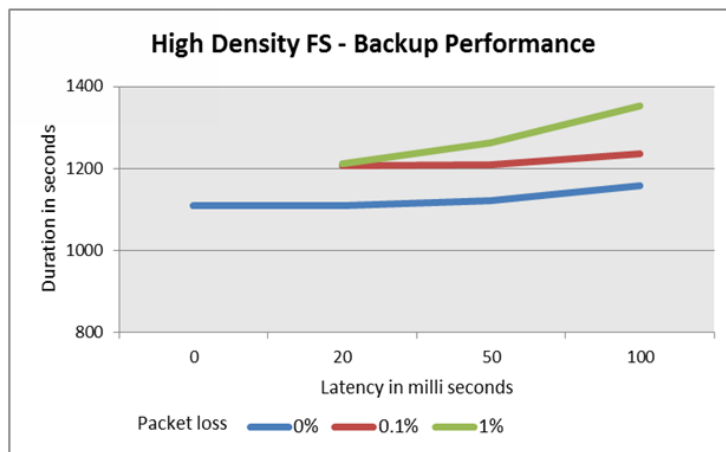


Figure 17 High Density FS Backup Performance - WAN between NetWorker Clients and Server

Observations and recommendations:

- NetWorker can sustain up to 200 ms latency with 0% packet loss between the NetWorker client and NetWorker server for metadata transfer of low density file systems.
- With a high density file system dataset (client backing up dataset with millions of files), the maximum latency between the NetWorker client and NetWorker server is 100 ms with 0% packet loss.
- NetWorker guarantees a 100% backup success rate with high latency on control path but any packet loss will significantly impact the backup success rate.
- If there is high latency link with packet loss between the NetWorker server and NetWorker client, EMC recommends setting higher client retry values in order for the backup to succeed. It was observed that with default client retries a backup with 200 ms and up to 5% packet loss succeeded.
- You will observe a 5–10% impact on the backup window when there is a high latency link between the NetWorker server and NetWorker clients.
- High packet loss (>5%) and high latency (>200 ms) results in a significant impact on the backup success rate and produce intermittent failures.

WAN Latency impact in data path (NetWorker clients and target device such as Data Domain separated by high latency) for backup

Figure 18 Large Density FS Backup Performance - WAN between NetWorker Clients and Data Domain

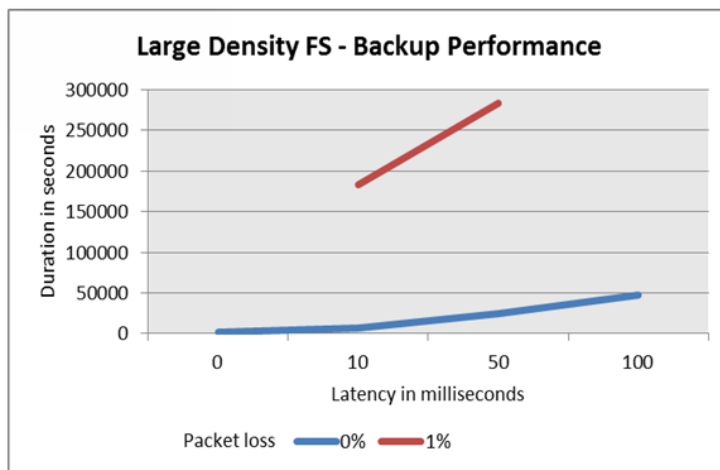
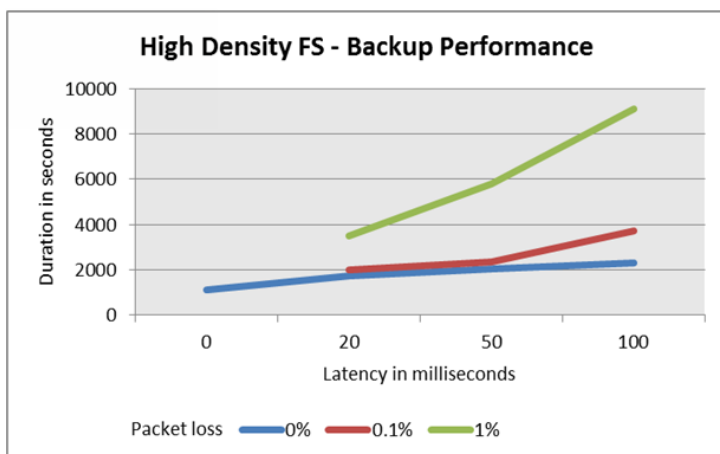


Figure 19 High Density FS Backup Performance - WAN between NetWorker Clients and Data Domain



Observations and recommendations:

- There is huge impact on throughput when there is high latency between the NetWorker client and Data Domain over WAN with client direct enabled.
- Every increase of 5 ms latency will slow the backup throughput by 2-2.5 times for the initial full backup to Data Domain over Boost. Higher latency link results significant slowness during backup.
- The consecutive backups (full or incremental) with very minimal change rate over high latency link impacts around 5-10% on backup throughput.
- EMC does not recommend exceeding the max latency of 50 ms between NetWorker client and remote Data Domain with very minimal (recommended 0%) packet loss.
- Packet loss in the data path has the most impact on the backup window and throughput. Even 1% of packet loss can reduce the backup throughput by 5-25 times, which significantly impacts the backup window.

WAN latency impact in data path (Latency between source and target DDR) for cloning

Figure 20 Large Density FS Clone Performance - WAN between NetWorker Clients and Data Domain

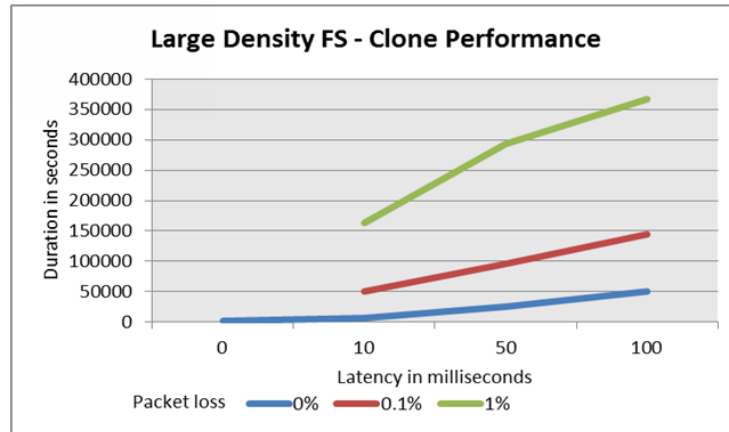
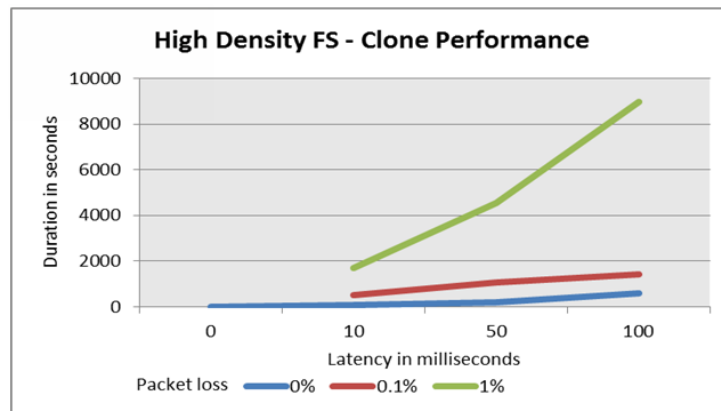


Figure 21 High Density FS Clone Performance - WAN between NetWorker Clients and Data Domain



Note

Clone-controlled replication (CCR) performance completely depends on the Data Domain model, the existing load on DDRs and the latency between two different Data Domain systems that are separated by WAN. The preceding results show the WAN latency impact between a DD880 (local site) and DD670 (remote site).

Observations and recommendations:

- If there is high latency link between the source and target DDR, then there is significant impact in clone throughput.
- Every 10 ms increase in latency reduces the clone throughput by 4-45 times.
- The packet loss in the WAN link further reduces the clone throughput by 4-300 times for a large density dataset and by 4-500 times for high density datasets.
- EMC does not recommend exceeding the 50 ms latency for large size datasets and 20 ms latency for high density dataset cloning.

WAN latency impact in data path (Latency between source and target DDR) for recovery

Figure 22 Large Density FS Recovery Performance - WAN between NetWorker Clients and Data Domain

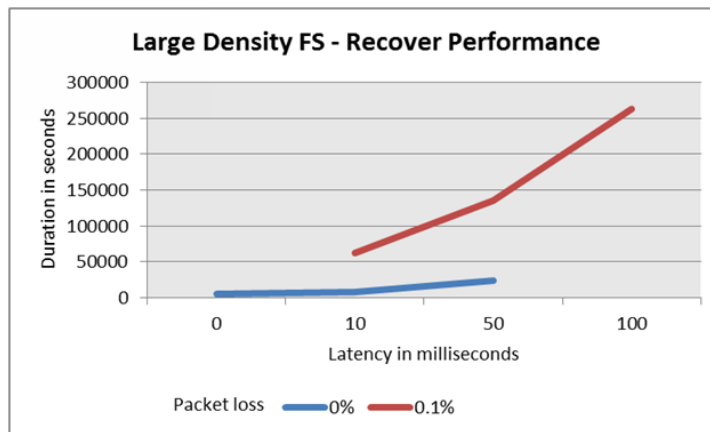
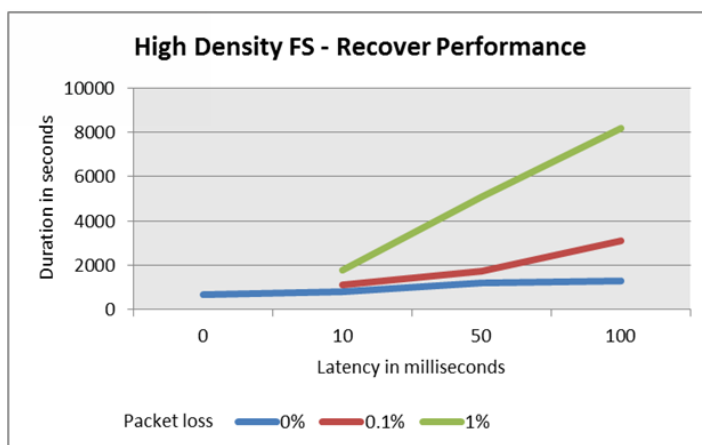


Figure 23 High Density FS Recovery Performance - WAN between NetWorker Clients and Data Domain



Note

For large density file system and high density file system dataset restore, time indicates the time that is taken to perform 10 simultaneous restores.

Observations and recommendations:

- Latency impacts recovery performance similar to the backup and clone workflows.
- If a high latency link exists between the NetWorker client and DDR during recovery, then the performance slows down drastically.
- Every 10 ms increase in latency reduces the recover throughput by 1-2 times for a high density dataset with multiple client restore. For a large density dataset with multiple client restore, throughput decreases by 2-10 times with increase in latency.
- The packet loss in the WAN link further reduces the restore throughput by 2-12 times.
- EMC does not recommend exceeding the 50 ms latency (with multiple restores) for a high dataset and 20 ms latency (with multiple restores) for a large dataset during recovery.

Summary

Table 10 Tolerable Range for Low Density file system

WAN path	Latency	Packet loss
Client - NetWorker server	0-100ms	0-1%
Client - Data Domain (DFA)	0-50ms	0-0.1%

Table 11 Tolerable Range for High Density file system

WAN path	Latency	Packet loss
Client - NetWorker server	0-100ms	0-0.1%
Client - Data Domain (DFA)	0-50ms	0-0.1%

Note

Higher latency and packet loss between the data path impacts throughput significantly. You can still use the high latency link for data path but the NetWorker server might re-attempt the failed backups due to packet loss. EMC recommends applying the preceding recommendations to avoid the failures with high latency WAN links.
