

Novell NetWare® 6.5

6.5

www.novell.com

SERVER MEMORY ADMINISTRATION
GUIDE



Novell®

Legal Notices

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

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

You may not export or re-export this product in violation of any applicable laws or regulations including, without limitation, U.S. export regulations or the laws of the country in which you reside.

Copyright © 2003 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

U.S. Patent No. 5,157,663; 5,349,642; 5,455,932; 5,553,139; 5,553,143; 5,572,528; 5,594,863; 5,608,903; 5,633,931; 5,652,854; 5,671,414; 5,677,851; 5,692,129; 5,701,459; 5,717,912; 5,758,069; 5,758,344; 5,781,724; 5,781,733; 5,784,560; 5,787,439; 5,818,936; 5,828,882; 5,832,274; 5,832,275; 5,832,483; 5,832,487; 5,859,978; 5,870,561; 5,870,739; 5,873,079; 5,878,415; 5,884,304; 5,893,118; 5,903,650; 5,903,720; 5,905,860; 5,910,803; 5,913,025; 5,913,209; 5,915,253; 5,925,108; 5,933,503; 5,933,826; 5,946,002; 5,946,467; 5,956,718; 5,956,745; 5,964,872; 5,974,474; 5,983,223; 5,983,234; 5,987,471; 5,991,810; 6,002,398; 6,014,667; 6,016,499; 6,023,586; 6,029,247; 6,052,724; 6,061,726; 6,061,740; 6,061,743; 6,065,017; 6,081,774; 6,081,814; 6,094,672; 6,098,090; 6,105,062; 6,105,069; 6,105,132; 6,115,039; 6,119,122; 6,144,959; 6,151,688; 6,157,925; 6,167,393; 6,173,289; 6,216,123; 6,219,652; 6,233,859; 6,247,149; 6,269,391; 6,286,010; 6,308,181; 6,314,520; 6,324,670; 6,338,112; 6,345,266; 6,353,898; 6,424,976; 6,466,944; 6,477,583; 6,477,648; 6,484,186; 6,496,865; 6,510,450; 6,516,325; 6,519,610; 6,532,451; 6,532,491; 6,539,381; RE37,178. Patents Pending.

Novell, Inc.
1800 South Novell Place
Provo, UT 84606
U.S.A.

www.novell.com

Server Memory Administration Guide

April 2003

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

Novell Trademarks

Novell is a registered trademark of Novell, Inc. in the United States and other countries.

eDirectory is a trademark of Novell, Inc.

NetWare is a registered trademark of Novell, Inc. in the United States and other countries.

NetWare Loadable Module and NLM are trademarks of Novell, Inc.

Novell Storage Services is a trademark of Novell, Inc.

Third-Party Trademarks

All third-party products are the property of their respective owners.

Contents

- About This Guide** **7**

- 1 Overview of Server Memory** **9**
 - What's New 9
 - Introduction to NetWare Memory 9
 - Caching 10
 - Logical Memory Addressing 12
 - Protected Address Spaces 12
 - Protected Address Spaces and Logical Memory Addressing 13
 - Modules Not Allowed in Protected Address Spaces 13
 - Virtual Memory 14
 - What's Next 15

- 2 Managing Server Memory** **17**
 - Server Memory Management Tools 17
 - Managing Server Processes 18
 - Using Protected Address Spaces 18
 - Loading Protected Address Spaces 19
 - Unloading Protected Address Spaces 21
 - Viewing Information about Address Spaces 22
 - What's Next 23

- 3 Optimizing Server Memory** **25**
 - Assessing Server RAM 25
 - Traditional File System 25
 - Novell Storage Services 26
 - Ensuring That Devices Can Access Memory 26
 - Determining Which Processes Use Too Much Memory 26
 - Tuning Server Memory 27
 - Tuning File Cache 27
 - Tuning Directory Cache for the Traditional File System 31
 - Using Virtual Memory 33
 - Using Swap Files 35
 - Responding to Disk Thrashing 35
 - Optimizing Garbage Collection 36

About This Guide

This guide describes how memory works on a NetWare 6.5 server and how to optimize and manage that server memory. This guide is intended for network administrators and is divided into the following sections:

- ♦ [Chapter 1, “Overview of Server Memory,”](#) on page 9
- ♦ [Chapter 2, “Managing Server Memory,”](#) on page 17
- ♦ [Chapter 3, “Optimizing Server Memory,”](#) on page 25

Additional Documentation

For more documentation on managing server memory, see the [NetWare Remote Manager Administration Guide](#).

Documentation Updates

For the most recent version of the *Server Memory Administration Guide*, see the [NetWare 6.5 Documentation Web site \(http://www.novell.com/documentation/lg/nw65\)](http://www.novell.com/documentation/lg/nw65).

Documentation Conventions

In this documentation, a greater-than symbol (>) is used to separate actions within a step and items within a cross-reference path.

A trademark symbol (® , ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party trademark.

1

Overview of Server Memory

The NetWare® server operating system provides increased memory efficiency with three important features: logical memory addressing, virtual memory, and protected memory. This section is a general introduction to these and other aspects of NetWare's memory architecture.

To manage memory and virtual memory, you need to understand the following:

- ◆ [“Introduction to NetWare Memory” on page 9](#)
- ◆ [“Logical Memory Addressing” on page 12](#)
- ◆ [“Protected Address Spaces” on page 12](#)
- ◆ [“Virtual Memory” on page 14](#)

What's New

NetWare 6.5 includes the following improvements for server memory:

- ◆ Memory moves quicker from one cache pool to another.
- ◆ Improved NetWare Remote Manager tools for reviewing memory.

For more information, see [Viewing Memory Configurations](#) in the [NetWare Remote Manager Administration Guide](#).

- ◆ Improvements in the way the server processes work.

For more information, see [“Managing Server Processes” on page 18](#).

Introduction to NetWare Memory

NetWare 6.5 requires a minimum of 512 MB of RAM and can access up to 64 GB of RAM. After the server.exe file is loaded, the remaining memory below 4 GB is assigned to **cache memory** and virtual memory, which is available for NetWare Loadable Module™ (NLM™) programs and other processes to use.

NOTE: The maximum memory that your server requires depends on the products and components that you install on your server and the number of applications that you run on your server that use protected memory.

The memory allocator can not allocate memory above 4 GB. If you install more than 4 GB, this memory is used by the virtual memory system, NSS, and applications such as eDirectory. The traditional file system does not use memory above 4 GB.

The applications that run in the memory above 4 GB leave more memory below 4 GB for the memory allocator to use. So while you cannot configure the memory above 4 GB, adding more memory has the following performance advantages:

- ◆ More memory is available to those systems and applications that run in protected memory or in the space above 4 GB.

- ◆ If the memory allocator has more available memory below 4 GB to use, then the faster the server performs.

Caching

A cache is a temporary, quickly accessible area in RAM that stores frequently used information from the hard disk. NetWare uses caching to increase the speed of reads and writes on server hardware. As long as a block of data stays in RAM, NetWare can access it much faster than accessing data on the disk.

A *cache hit* means that the server found the data in RAM rather than having to go to disk. This provides a large performance increase.

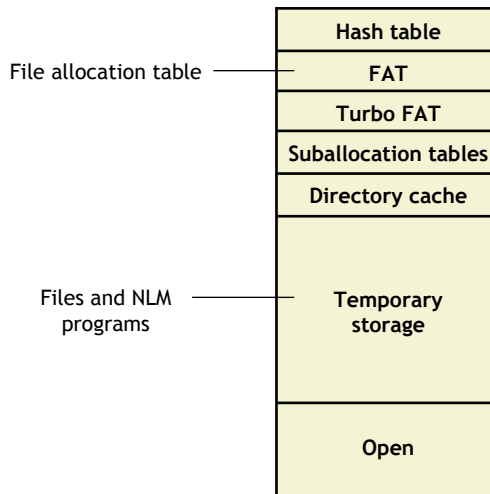
Having sufficient RAM to cache disk data results in a high percentage of cache hits. The more NLMS and other services you load, the less RAM the server has available for caching data. This lowers your cache hit percentage and slows network performance. For information on disk cache utilization, see [“Tuning Server Memory” on page 27](#).

When data is stored in cache, it is stored in 4 KB blocks called cache buffers. Recently used data is kept in cache buffers in case it is needed again.

Traditional File System

Cache memory is used for all the server's processing needs, such as the following:

- ◆ Storing volumes' File Allocation Tables (FAT) and suballocation tables
- ◆ Storing volumes' directory entry tables
- ◆ Storing data files for users
- ◆ Storing NLM programs, such as LAN drivers, disk drivers, and management utilities
- ◆ Building hash tables for directory names
- ◆ Creating protected address spaces
- ◆ Allocated memory used by NLM programs
- ◆ Name cache and file cache



Memory from cache is allocated to an NLM program when requested. When an NLM program is finished using the memory, it releases the memory and returns it to cache. NetWare also provides a **garbage collection** process to recover unused memory.

Novell Storage Services

In NetWare 6.5, Novell Storage Services (NSS) is always loaded on the server in order to run and cache memory is always allocated. NSS uses memory differently than the traditional file system. Some of these differences are the following:

- ◆ NSS caches files larger than 128K in the memory above 4 GB. This improve performances for these files and for smaller files because they do not compete for the same memory.
- ◆ NSS has its own cache blocks, although they are counted as total free blocks in the system. The memory allocator can use them for applications other than NSS.
- ◆ By default, NSS maintains the Closed File Cache Size parameter at 100,000. Each cache is about 1K, so 100 MB of memory is eventually dedicated to Closed File Cache Size. NSS does not automatically allocate 100 MB of memory for this parameter; it waits for 100,000 files to close before it reaches 100 MB.

If the server detects that it needs more memory, some of this 100 MB will automatically be sent to the free pools. NSS adjusts itself as pressure on the memory system increase. For more information about this parameter, see **“Cache Hit Percentage” on page 28**.

You must use specific parameters for tuning your server when using NSS. For more information, see **“Tuning File Cache” on page 27**.

Directory Buffers and File Cache Buffers

There are two kinds of cache: directory cache and file cache.

Directory cache is used only with the traditional NetWare file system, not with the Novell® Storage Services™ (NSS) file system. The directory cache is a percentage of the file cache assigned to the traditional file system. Directory cache stores disk directory entries. When the server is first installed, NetWare allocates 20 directory cache buffers of 4 KB each. If more buffers are needed, NetWare tunes the server by increasing the number of directory cache buffers.

You can tune the number of directory cache buffers yourself, using two server parameters that establish the minimum and maximum number of directory cache buffers the server can allocate. For information about tuning directory cache, see **“Tuning Directory Cache for the Traditional File System” on page 31**.

File cache is used with both the traditional file system and the NSS file system. File cache is a pool of 4 KB memory pages that stores recently used file data.

In an ideal environment, there is enough memory so that all frequently used data can be retrieved from file cache rather than from disk.

Monitoring the disk cache utilization statistic can help you estimate whether your server has sufficient memory to service requests from cache. For information about optimizing file cache, see **“Tuning File Cache” on page 27**.

If you are using both Novell Storage Services™ (NSS) and the traditional file system, you want to tune both the file and directory caches. You should only tune the directory cache if you are using the traditional file system.

Garbage Collection

NetWare uses memory deallocation and garbage collection to collect previously allocated but currently unused segments of memory and return them to a common memory pool. Garbage collection is used by the traditional file system only.

Garbage collection is triggered by demand on the virtual memory system. The virtual memory system is checked every 10 seconds. For an explanation of virtual memory, see [“Virtual Memory” on page 14](#).

If virtual memory is being used heavily, garbage collection happens immediately. If virtual memory usage is low, garbage collection happens according to the following process:

1. A process named Free labels the memory as deallocated.
2. The garbage collection process recovers the pieces of deallocated memory and returns them to the memory pool.

The garbage collection is interruptible, can run in the background, and should run frequently. You can manually initiate the process if needed. See [“Optimizing Garbage Collection” on page 36](#).

Logical Memory Addressing

NetWare’s logical memory addressing scheme helps to prevent memory fragmentation. The physical memory of the server is divided into 4 KB blocks called pages. When a NLM program wants to store information in memory, it requests memory pages that are contiguous, so that all the information can be stored in a sequential group of pages.

As contiguous blocks of memory pages are allocated and then made free, small segments of memory are often left over. After a while, there are many such isolated pieces of memory too small to be used. This situation is called memory fragmentation. When memory is fragmented, the server often cannot satisfy memory requests even when a lot of memory is available.

The memory management subsystem provides a logical space manager that lists logical memory addresses from 0 to 4 GB. The logical space manager lists addresses up to 4 GB even if the server contains only the minimum 256 MB of RAM.

When an NLM program needs a contiguous block of memory, the operating system finds a set of logical, contiguous addresses in the logical space. Because the logical space has so many logical addresses (one million entries—one for each 4 KB page), the operating system can almost always find the contiguous logical memory addresses the NLM program needs. Then the operating system allocates physical memory, which might or might not be contiguous. The operating system then binds the logical addresses to the physical addresses.

Protected Address Spaces

A portion of the logical address space is set aside for protected address spaces (sometimes called user address spaces or ring 3). This protected address space is a portion of the logical address space that has carefully controlled interaction with the server operating system. All protected address spaces use virtual memory. For general information about virtual memory, see [“Virtual Memory” on page 14](#).

A key benefit of protected address spaces is that you can use them to take advantage of virtual memory or to run untried or troublesome applications. Because modules loaded into a protected address space can’t corrupt the operating system or cause server abends, the protected address space provides a safe place to run applications.

The operating system address space (ring 0) is sometimes called the OS address space or the kernel address space. The operating system itself cannot run in a protected address space.

The memory protection marshalling layer, in conjunction with the memory protection subsystem, prevents modules in a protected address space from having direct access to anything outside the address space.

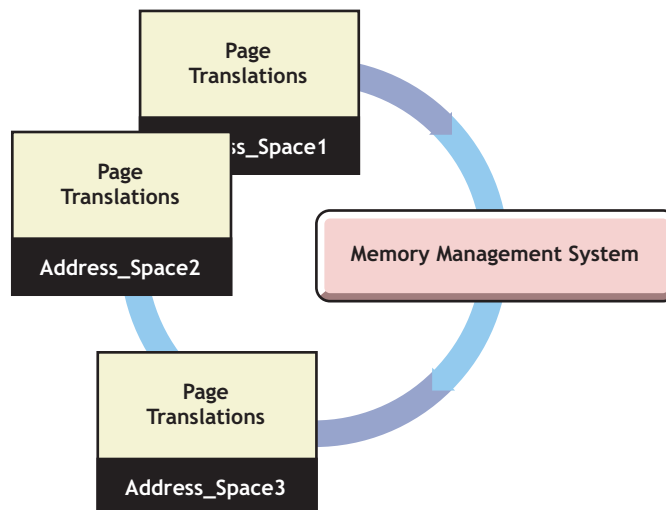
In particular, the memory protection marshalling layer serves as the interface between the protected address spaces and the server operating system. These layers prevent NLM programs in protected spaces from making calls or passing parameters to the operating system that would corrupt or fault the core operating system. Likewise, programs using memory outside the boundary cannot allocate areas of memory inside the boundary.

You can load modules into a protected address space, unload modules from a space, delete a space, or kill a space. See [“Using Protected Address Spaces” on page 18](#) and [“Loading Protected Address Spaces” on page 19](#).

Protected Address Spaces and Logical Memory Addressing

Each protected address space has its own set of page translations to provide logical memory addressing. The memory management subsystem ensures that the page translations of each protected address space refers to a different set of physical memory addresses.

Only one set of translations is visible in the memory management subsystem at a time. When there is more than one protected address space, the memory management subsystem loads one set of page translations for a specified time and then replaces it with another.



Replacing visible page translations is part of a context switch, which takes place when the NetWare kernel changes from running one thread to running another thread.

Modules Not Allowed in Protected Address Spaces

The following NLM programs and executable files cannot be loaded into a protected space:

- ♦ Server.exe
- ♦ LAN and disk drivers

Hardware and LAN drivers make calls to the operating system and the processor that are not allowed from the protected address spaces.

- ◆ Monitor

Monitor makes calls to the operating system that are not allowed from protected address spaces.

- ◆ Modules that use interrupt time processing
- ◆ Modules that use direct operating system interfaces
- ◆ Modules that import operating system data items
- ◆ Other modules or applications that aren't designed to run in protected address spaces

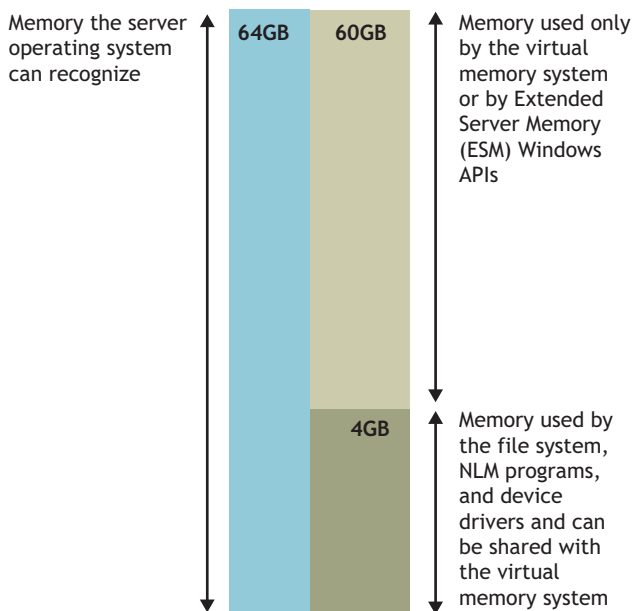
Virtual Memory

NetWare also provides a virtual memory system that swaps little-used data to disk, thereby freeing memory for more frequently used data. Virtual memory provides more efficient use of memory and lessens the likelihood that low memory conditions will cause problems with the server.

Applications that run in protected address spaces use virtual memory, as does Java*. Applications running in protected space can allocate more memory than if they were not in protected space. Running in protected space keeps an application from corrupting other applications or the kernel; the application is also protected from other applications.

Keep in mind that while virtual memory can be safer for applications and increases the amount of memory that can be used on the server, it also runs slightly slower.

The following illustration shows the amount memory the server can recognize and what portions the virtual memory system uses.



Server operating system modules do not use virtual memory because of the nonpreemptive nature of the NetWare kernel.

NOTE: If an NLM program was designed to use virtual memory and it implemented the needed locking schemes, a NetWare kernel NLM program could be written to use virtual memory.

A periodic aging process assesses available memory to see which data has been used less recently than the rest. Data that has not been used for some time can be moved from memory to disk, thereby freeing memory for other uses.

When the data on disk is needed again, it is moved back into available memory. Because data is swapped in and out of the disk, the available memory can be used for a larger amount of data than its actual physical capacity would allow.

The memory management subsystem assesses the memory needs of all server and NLM processes and determines where the freed memory should be used.

The Java Virtual Machine and any modules that are loaded into protected address spaces use virtual memory. Both the modules and the data they access are subject to being swapped to disk.

When data is moved from memory to disk, it is stored in a swap file. You can create one swap file for each NetWare volume. A swap file is created for volume sys: by default.

To create a swap file, delete a swap file, change swap file parameters, or view information about the swap file, you can use NetWare Remote Manager, Monitor, or the **swap** command at the System Console prompt. See [“Using Virtual Memory” on page 33](#).

Swapping the data between memory and disk requires additional system resources or CPU time, but it increases the memory available for use. The availability of the additional memory allows more NLM programs to run in the same amount of memory but they will probably run more slowly. Overall server performance might improve but probably won't.

What's Next

For information on managing server memory, begin with [“Server Memory Management Tools” on page 17](#).

To increase the efficiency of server memory, see [Chapter 3, “Optimizing Server Memory,” on page 25](#).

2

Managing Server Memory

This section briefly describes the tools you use to manage memory, and how you can manage server processes, and how to protect your operating system by using protected address spaces.

- ◆ “[Server Memory Management Tools](#)” on page 17
- ◆ “[Managing Server Processes](#)” on page 18
- ◆ “[Using Protected Address Spaces](#)” on page 18
- ◆ “[Loading Protected Address Spaces](#)” on page 19
- ◆ “[Unloading Protected Address Spaces](#)” on page 21
- ◆ “[Viewing Information about Address Spaces](#)” on page 22

Server Memory Management Tools

NetWare® provides several tools to help you manage and tune the memory system:

- ◆ NetWare Remote Manager (portal.nlm) provides the ability to monitor the memory used in each of the following categories:
 - ◆ NLM Memory
 - ◆ File System Cache
 - ◆ Memory Used by the Virtual Memory System: Swap File Information, Virtual Memory Cache Pool Information, and Virtual Memory Statistics
 - ◆ Available Cache Memory
 - ◆ DOS, Video, and Other Memory
 - ◆ Total System Memory

You can also use NetWare Remote Manager to load and unload modules into protected address spaces, change set parameter values, access the server console to run Monitor, and execute any console commands.

In most cases, NetWare Remote Manager is more useful than Monitor when diagnosing server memory problems. For more information, see the *NetWare Remote Manager Administration Guide*.

- ◆ The Monitor utility provides statistics that indicate how efficiently memory is being used by the server and individual modules. For more information, see [Monitor](#) in the *Utilities Reference*.
- ◆ Settable parameters and console commands let you tune the memory system to achieve optimal performance. For more information, see [Set > Memory Parameters](#) in the *Utilities Reference*.
- ◆ For NSS, you can enter commands at the system console or use NetWare Remote Manager.

Managing Server Processes

In NetWare 6.5, the following changes were made in the way that the server processes work:

- ◆ Eliminated the New Service Process Wait Time parameter.
- ◆ Allowed the kernel to allocate up to the Maximum Server Processes when they are needed without waiting.
- ◆ Freed worker threads that haven't been used for five minutes and retained Minimum Server Processes.
- ◆ Server process names won't be reused, so higher numbers than the current number of worker threads will be available.

These changes enable you to configure the Maximum Server Processes to a higher level without worrying that memory will be wasted by unused server processes.

A good rule is to have your Maximum Server Processes set to a server process for each remote client that will be logged in to the server.

You can configure Maximum and Minimum Server Processes in Netware Remote Manager. On the Set Parameters page, click Miscellaneous and then click either Maximum or Minimum Service Processes.

You can see if you are near the configured maximum for either pool in Monitor, which now reports both the current MP server processes and legacy UP server processes. To see if you are having packet receive buffer problems:

- ◆ In NetWare Remote Manager, on the LSL Statistical Information page look at the Get ECB Fails statistic.
- ◆ In Monitor, on the LAN Statistics screen look at Receive Discarded, No Available Buffers statistic.

If you are running out of packet receive buffers, you might not have enough of them or you might need more server processes. First increase the number of packet receive buffers. If that doesn't resolve the problem, increase the number of server processes.

Note that packet receive buffers are usually 4 KB in size and each server process requires slightly more than 32 KB of memory.

Using Protected Address Spaces

You can use a portion of server memory called a protected address space to protect the server operating system from untried or troublesome applications and to use virtual memory. When applications are loaded into a protected address space, they can't corrupt the operating system, cause server abends, or corrupt other applications that are running in other protected address spaces.

All protected address spaces use virtual memory, so running modules in a protected address space also uses RAM more efficiently.

For general information about protected address spaces and the applications that can be loaded into them, see [“Protected Address Spaces” on page 12](#). For general information about virtual memory, see [“Virtual Memory” on page 14](#).

- ◆ Because modules loaded into a protected address space have controlled interaction with the operating system, all modules that must communicate with each other by directly importing

another module's data or functions (such as those of an e-mail application or a database) should be loaded into the same protected address space.

- ◆ If you load a NetWare Loadable Module™ (NLM™) that depends upon another module, such as CLIB, the other module will be loaded automatically into the same address space.
- ◆ If the same module is loaded into more than one address space, the module's code will be shared among the address spaces. Therefore, loading the module into multiple address spaces does not require additional memory for the module itself. Only data for the required module is unique for each address space.
- ◆ Even if an NLM is designed to be loaded only once, you can load multiple copies of the NLM on your server if you load them into different address spaces.
- ◆ When you load modules into a protected address space, NetWare assigns whatever amount of memory the loaded modules need, up to a maximum size of 512 MB. The maximum size of a protected address space is fixed; but within that maximum limit, the memory size of the space grows and shrinks as needed by the modules in the address space.
- ◆ If you want the server to clean up address spaces that abend, in NetWare Remote Manager make sure the Memory Protection Fault Cleanup parameter (Memory category) is set to the default setting of On. If an address space abends, the server removes the address space and its modules and returns the resources to the system.

If the Memory Protection Fault Cleanup parameter is set to Off, the situation is left to the NetWare kernel's abend recovery mechanism.

Loading Protected Address Spaces

You can create a protected address space when you load one or more modules.

To load protected address spaces, use the commands in the following table at the System Console prompt or in NetWare Remote Manager on the Address Space Information page (accessed by clicking Protected Memory in the navigation frame).

To	Enter this command at the System Console prompt	Do this on the Address Space Information Page of NetWare Remote Manager	Example
Load one module into the next available protected address space	<code>[load] protected module_name</code>	<ol style="list-style-type: none"> 1. Type <code>module_name</code> in the Load Module Protected field. 2. Click Load Module Protected. 	<p><code>[load] protected database.nlm</code></p> <p>Creates a new address space called <code>address_spacen</code>, where <code>n</code> is a numerical number, and loads <code>database.nlm</code> into it.</p>

To	Enter this command at the System Console prompt	Do this on the Address Space Information Page of NetWare Remote Manager	Example
<p>Load one module into the next available protected address space with restart functionality</p> <ul style="list-style-type: none"> Restart functionality means that if the protected space abends, the system closes the space, cleans up its resources, restarts the space, and reloads the module into it. To prevent the server from restarting a memory space that continues to fault, use the Memory Protection No Restart Interval Set parameter (Memory category). 	<pre>[load] restart module_name</pre>	<ol style="list-style-type: none"> Type restart <i>module_name</i> in the Load Module Protected field. Click Load Module Protected. (Optional) If you want to change the restart state to No, click Yes in the Restartable column of the Address Space Information table. 	<pre>restart database.nlm</pre> <p>Creates a new address space called <i>address_space_n</i>, where <i>n</i> is a numerical number, and loads <i>database.nlm</i> into it.</p> <p>If the address space abends, the system will shut down the space, restart the space automatically, and reload <i>database.nlm</i> into it.</p>
<p>Load one module into a protected space with a user-defined name</p> <ul style="list-style-type: none"> Use this command when you want to create your own name for the space. Also use this command when you want to load more than one module into the same protected space. Repeat the command for each module you want to load into the space. You can also include the restart option on the same command line. 	<pre>[load] address space = space_name module_name</pre>	<ol style="list-style-type: none"> Type address <i>space = space_name module_name</i> in the Load Module Protected field. Click Load Module Protected. (Optional) If you want to load more than one module in the same space, repeat Steps 1 and 2 for each module you want to load. 	<pre>[load] address space = db_space database.nlm</pre> <p>Creates a new address space called <i>db_space</i> and loads <i>database.nlm</i> into it.</p> <pre>[load] address_space = db_space restart test.nlm</pre> <p>Loads <i>test.nlm</i> into the <i>db_space</i> address space and also adds restart functionality to the address space.</p> <pre>[load] address space = db_space test.nlm</pre> <p>Loads the <i>test.nlm</i> in to the same <i>db_space</i> address space that the <i>database.nlm</i> was loaded into.</p>

To	Enter this command at the System Console prompt	Do this on the Address Space Information Page of NetWare Remote Manager	Example
<p>Load multiple modules at once using an .ncf file</p> <p>The command creates a protected space with the same name as the .ncf file and executes the .ncf file to load all the modules into the space.</p>	<code>protect filename</code>	<ol style="list-style-type: none"> Type <i>NCF_filename</i> in the Load NCF File Protected field. Click Load NCF File Protected. 	<p><code>protect load_db</code></p> <p>Creates an address space called <code>load_db</code> and executes <code>load_db.ncf</code> to load modules into the protected space.</p>

Unloading Protected Address Spaces

To unload a module from a protected space or to unload the space itself, use the commands in the following table at the System Console prompt or in NetWare Remote Manager on the Address Space Information table (accessed by clicking Protected Memory in the navigation frame).

To	Enter this command at the System Console prompt	Do this on the Address Space Information Page of NetWare Remote Manager	Example
<p>Unload all modules from the designated address space and removes the space</p>	<code>unload address space = space_name</code>	Click Unload in the Unload column of the Address Space Information table for the address you want to unload the modules from.	<p><code>unload address space = address_space1</code></p> <p>Unloads all the modules from <code>address_space1</code>, closes <code>address_space1</code>, and returns its resources to the system.</p>
<p>Remove an address space without unloading the modules first</p> <p>Use this command only if you know the space can't be unloaded with the <code>unload address space</code> command.</p>	<code>unload kill address space = space_name</code>	Click Remove in the Remove column of the Address Space Information table for the address space you want to remove.	<p><code>unload kill address space=address_space1</code></p> <p>Removes <code>address_space1</code> without unloading modules and returns its resources to the system.</p>

If a module hangs while it is being unloaded with the `unload address space` command, the system waits a specified time and then displays the following error message:

```
Module_name in address_space did not unload yet. Killing the address space is the only way to remove the NLM. Should the address space be killed? Yes/No.
```

If you answer Yes, the system shuts down the address space without unloading modules from it. If you answer No, the system waits a specified time and displays the message again.

You can designate the amount of time before the message appears by setting the Hung Unload Wait Delay (Error Handling category) parameter. The default is 30 seconds.

In some circumstances, the server might not completely clean-up an address space that has faulted. This failure is caused by NLM programs that are not written to allow external cancellation.

When NetWare cannot clean up an address space after it faults, it sends an alert message to the server console. If you execute the **protection** command, the address space will be listed as in the clean-up failure state.

Because you cannot unload an address space that is in the clean-up failure state, you must shut down and restart the server to recover the remaining resources from the address space.

Viewing Information about Address Spaces

You can see which modules are loaded in which address spaces.

In NetWare Remote Manager

- 1 Click Protected Memory in the navigation frame.
- 2 View the address space information on the Protected Memory page.
- 3 To view the modules that are loaded in a specific address space, click Address Space *Name* for the specific address.

At the System Console Prompt

- 1 To display a list of loaded modules with their address space names, enter

```
modules
```

Each NLM name displayed is followed by the name of the address space where the NLM resides, the full utility name, version number, date, and other information. The module names are displayed in the order in which they were loaded.

- 2 To display a list of all address spaces, enter

```
protection
```

Each address space name that is displayed is followed by a list of all modules loaded into that space, with a short description of each module.

In Monitor

To view additional statistics for address spaces, do the following:

- 1 Click Available Options > Virtual Memory > Address Spaces.
- 2 Select an address space.
 - ♦ The information window displays data about the selected address space.
 - ♦ To activate and expand the information window, press Tab.
 - ♦ To see a description of each field, press F1.
 - ♦ To exit the window, press Esc twice.
- 3 Select a specific address space and press Enter to see a list of all modules loaded into the address space.
 - ♦ Select a module in the list.

The information window displays data about the selected module
 - ♦ To activate and expand the information window, press Tab.

- ◆ To see a description of each field, press F1.
- ◆ To exit the window, press Esc twice.

What's Next

To increase the efficiency of server memory, see [Chapter 3, “Optimizing Server Memory,”](#) on page 25.

3

Optimizing Server Memory

This section identifies tasks that you can perform to optimize server memory and thus increase network performance. Optimizing server memory includes the following tasks:

- ♦ “Assessing Server RAM” on page 25
- ♦ “Ensuring That Devices Can Access Memory” on page 26
- ♦ “Determining Which Processes Use Too Much Memory” on page 26
- ♦ “Tuning Server Memory” on page 27
- ♦ “Using Virtual Memory” on page 33
- ♦ “Optimizing Garbage Collection” on page 36

Assessing Server RAM

Insufficient physical memory (RAM) in the server is a common cause of a slow network. To determine whether your server has sufficient physical memory installed, check the amount of server memory currently available for disk cache.

Traditional File System

To check the amount of server memory currently available for disk cache, you can use either NetWare® Remote Manager or Monitor.

In NetWare Remote Manager

- 1** Click Health Monitor in the navigation frame, then view the available memory status and values.
- 2** If the status is Suspect or Bad, view the View Memory Config page to determine the source of the problem. To do this, click Available Memory on the Health Monitor page or click View Memory Config in the navigation frame.

In Monitor

- 1** View the General Information window and note the percentage for the Long Term Cache Hits.
- 2** If the percentage of Long Term Cache Hits falls below 90 percent, add more RAM to the server.

HINT: If you are unable to add more RAM now, a short-term fix can help. We suggest unloading unneeded NLM™ programs or removing DOS from memory. Remember that these actions treat symptoms only and are not long-term solutions.

Novell Storage Services

To check the amount of server memory used for disk cache, use the following commands through NetWare Remote Manager or at the system console:

- ♦ **nss /status**
Reports on general caching parameters for NSS.
- ♦ **nss /cachestats**
Displays statistics for cache buffers used by NSS.
- ♦ **nss /resetstats**
Resets the cache stats. Use this command to trace statistics from day to day, especially during the busy hours. Keep in mind that running backup will alter the statistics.
- ♦ **nss /namecachestats**
Reports on name cache, including file names and hit rates.

Ensuring That Devices Can Access Memory

Some older network boards, such as many ISA and MCA devices, cannot access memory above 16 MB. To make sure these devices have enough memory below 16 MB, make sure that the following **set** command is in the startup.ncf file:

```
set reserved buffers below 16 meg = 300
```

The default number of buffers is 300. If necessary, you can increase the value, but remember that too many reserved buffers can prevent large volumes from mounting. As soon as possible, upgrade the system to a newer board that can access higher memory addresses. For a discussion of network bottlenecks, see [Matching Performance Capabilities](#) in the *Server Operating System Administration Guide*.

NetWare automatically registers memory above 16 MB so the memory can be recognized by EISA and PCI machines. If memory above 16 MB isn't being recognized, you can register memory manually using the **register memory** command. For more information, see [Register Memory](#) in the *Utilities Reference*.

IMPORTANT: Use the **register memory** command only if absolutely necessary. Manually registering memory can cause memory fragmentation problems. Ideally, you should upgrade the system board so that the NetWare automatic memory registration will work.

Determining Which Processes Use Too Much Memory

To determine which NLM programs are using excessive amounts of memory (have a memory leak), complete the following steps.

NOTE: Complete [Step 1](#) in either NetWare Remote Manager or Monitor.

- 1** Watch the memory usage trends for any NLM that you might suspect has a problem.
Any NLM that has a memory pool that is constantly growing might have a memory leak.
- 2** Watch for "Short term memory allocator is out of memory" alerts displayed at the System Console or Logger screen or in the system log files.
Often one NLM stands out with an unusually large memory pool.

In NetWare Remote Manager

- 1 Click List Modules in the navigation frame.
- 2 Sort the list by allocated memory usage by clicking Alloc Memory in the table.
NOTE: Server.nlm and nss.nlm modules will always be high memory users.
- 3 View the Allocation Summary page for each NLM that is listed at the top of the list by clicking Alloc Memory for each NLM.

In Monitor

- 1 Click Available Options > Loaded Modules.
- 2 Select a module from the list to display its information and statistics in the upper window.
- 3 Press Tab to expand and activate the Module Information window.
- 4 Check the module's allocated memory.

The information includes bytes of memory required to load, allocated memory bytes in use, percent of memory in use, allocated memory bytes free, percent of memory free, and the module's load flags. Press F1 for a description of these statistics.

- 5 Press Tab to return to the list of loaded modules.
- 6 Check the memory statistics for other modules.
- 7 (Optional) Press F4 to free memory allocated for a selected module.

If you suspect an NLM program has a memory leak, you can unload and reload the NLM to solve the problem temporarily. To solve the problem permanently, contact the vendor of the program and request them to fix the program.

Tuning Server Memory

NetWare is a self-tuning operating system; however, there are some things you can do to adjust the server's memory subsystem (as well as other aspects of server operation) to fine tune its performance.

Before attempting to tune a new server's memory, let the server operate at full capacity for a week or two to allow NetWare to self-tune. After the server settles in to a predictable pattern of daily use, you can begin to fine-tune the server's memory as described in the following sections.

- ♦ [“Tuning File Cache” on page 27](#)
- ♦ [“Tuning Directory Cache for the Traditional File System” on page 31](#)

If you are using both Novell[®] Storage Services[™] (NSS) and the traditional file system, you want to tune both the file and directory caches.

Tuning File Cache

File cache speeds access to file data. If you want to tune your NetWare server in general, tune the file cache.

Novell Storage Services File System

In NetWare 6.5, NSS is always loaded on the server in order to run, and cache memory is always allocated. NSS uses memory differently than the traditional file system does, and you must use specific parameters for tuning your server when using NSS.

In cases where you have upgraded a server to NetWare 6.5 and you have no NSS logical volumes, you need to set the following NSS cache values as noted:

- ◆ **nss /noCacheBalance**
- ◆ **nss /MinBufferCacheSize=2048**

In NetWare 6.5, all CD and DVD drives are recognized as NSS devices; therefore, you need to set NSS cache parameters to accommodate the usage of these devices.

To view statistics for cache buffers used by NSS, enter one of the following commands at the System Console prompt:

```
nss /cachestats
```

or

```
nss /reset stats
```

For heavy access on an NSS logical volume, the server might incur some performance issues. NSS has its own caching and parameters to be set during the loading of the NSS modules.

Cache Hit Percentage

From an overall server performance perspective, the most important statistic to look at is cache hit percentage. Cache hit percentage indicates how often NSS looked in cache for a file and found it there. To view the cache hit percentage, use the **nss /cachestats** command.

The long term cache hit percentage should be above 90%. A long-term hit percentage below 90% can indicate you have too little RAM on the server, or that one or more of the following parameters is set too low.

- ◆ **Cachebalance** controls how much memory is available for NSS to use for cache. The cachebalance percentage determines how many cache blocks NSS will take from the traditional file system for its own cache. A high cache balance percentage will impede the performance of the traditional file system. A low cache balance will impede the performance of NSS.

We recommend that you set the cache balance parameter to equal the percentage of the total disk space you allocate for NSS. The default is 85 percent. If you are using mostly NSS volumes, you can run at a higher percentage, such as 90 percent.

- ◆ **Closefilecachesize** dictates how many closed files are kept in cache. The default is 100,000. If applications on a server consistently open and close many files, increase this setting. If the applications use only a few file large files, decrease this setting so more of memory can be used for the buffer cache.

NSS adjusts the number of closed files it keeps in cache to balance the needs of the file system with the needs of other processes. NSS does not automatically keep the number of closed files in cache that are indicated by this parameter. This parameter just indicates a maximum amount.

- ◆ **Fileflushtimer and Bufferflushtimer** set the number of seconds that modified files and buffers stay in memory before being written to disk. Increasing these parameters from their

defaults helps the caching up to a certain point, but too much increase causes a *throttling* problem with writing out the metadata. This is somewhat dependent on processor speed and disk speed, so you could try adjusting them for your system. If NSS performance is very poor, 75 might be too high and you could try 50. If performance is just below normal, you could try increasing the values a little.

To activate these parameters, you can issue the following command at the startup of NSS, or place it in the `autoexec.ncf` file:

```
nss /cachebalance=80 /fileflushtimer=75 /bufferflushtimer=75 /  
closedfilecachesize=50000
```

NSS caches files greater than 128K in the memory above 4 GB. This improves performance for these files and for smaller files because they won't be competing for the same space in the memory below 4 GB.

For more information on setting Cache Buffers for NSS, see [Setting the Cache Buffers](#) in the *Storage Services Administration Guide*.

Assessing Changes to NSS Settings

After making changes to NSS settings, enter the following command at the System Console prompt:

```
nss/cachestats
```

Note the statistics and how long the server has been up. After noting the statistics, enter the following command at the System Console prompt:

```
nss/resetstats
```

This command clears the file system cache statistics and starts keeping new statistics. This allows you to see the result of your changes.

Before you consider the results valid, you must allow NSS to keep statistics for at least two full business days. If the server has been up for more than a few weeks before you reset the statistics, wait at least a week before you consider the results valid.

Traditional File System

Because it is much faster to get data from memory than to get it from disk, caching frequently requested data increases server performance. The more RAM installed in the server, the more cache memory is available. Cache hits occur when a client requests data and the server supplies the data from cache memory instead of from disk. The **least recently used** algorithm determines whether data is kept in cache memory or written to disk.

File cache buffers are organized into a linked list. Most recently used (MRU) buffers are at the head of the list and least recently used (LRU) buffers are at the end of the list. The length of time the oldest buffer has been in the list is shown as the LRU Sitting Time statistic.

In tuning file cache for the traditional file system, the goal is to determine how much memory the server needs so that recently accessed data can always be retrieved from cache. Monitoring the LRU Sitting Time statistic is key to tuning file cache.

Understanding LRU Sitting Time

The LRU Sitting Time status is updated and displayed once per second. The statistic is calculated by taking the difference between the current time and the time stamp of the last block in the cache list. (The time stamp is the time when the last block entered the list.) The result is displayed in

HH:MM:SS.0 format. The LRU Sitting Time measures the length of time it takes for a cache buffer at the head of the list to make its way down to the tail, where it becomes the least recently used (LRU) buffer. Every cache buffer in the list is being reused within that period of time.

In configurations with excessive cache, the LRU Sitting Time can be very high, even many hours. In configurations with too little cache, the LRU Sitting Time can be measured in seconds. The time can vary widely, depending on circumstances.

On inactive servers, such as those sitting unused overnight or those in lab environments with long periods of idle time, the LRU Sitting Time statistic is incremented by one second every second. This is because the LRU Sitting Time indicates the age of the LRU cache buffer. The LRU Sitting Time statistic is useless under these circumstances, except to confirm the obvious: that data is not being read from or written to the server's cache. This statistic is most useful during peak workloads when the server's cache is servicing the greatest number of users and the broadest possible data set.

You can view the LRU Sitting Time statistic using NetWare Remote Manager or Monitor.

In NetWare Remote Manager:

- 1 Click View Memory Config in the navigation frame.
- 2 Click File System Cache on the View Memory Config page.

In Monitor:

- 1 Click Available Options, then select Disk Cache Utilization.

Calculating Cache Memory Needs

To determine how much file cache your server needs, do the following:

- 1 Track server resource utilization statistics.
Chart the results for daily, weekly, monthly, period-end, and year-end cycles. Identify recurring periods of peak workloads.
- 2 Observe the LRU Sitting Time statistic during peak workload periods of the day.
Keep a record of the *lowest* LRU Sitting Time during the peak periods for at least one week, longer if necessary, to see a consistent pattern.
- 3 Based on the observations you made of the server's resource utilization and LRU Sitting Time, determine an average low LRU Sitting Time.
- 4 Tune the cache.

Size the cache in such a way that the server can sustain an average low LRU Sitting Time of at least 12 minutes during peak workloads.

For example, if your server's original average low LRU was 7 minutes, you will need to add enough memory to increase the LRU Sitting Time to an average low of 12 minutes during peak workloads. This added memory increases the likelihood that repeatedly used data will still be cached when the next user request is received.

On the other hand, if the server's original lowest LRU sitting time was 18 minutes, the server has more than adequate cache. You can leave the excess memory in the server for future growth or remove it and use it somewhere else.

Setting a File Cache Buffer Alert

You can control when the server alerts you to serious decreases in the number of available file cache buffers. In NetWare Remote Manager, click Set Parameters > Traditional File System.

- ♦ **Minimum File Cache Buffers** establishes the minimum number of buffers that will always be reserved for file cache and not made available for other processes. When this minimum is reached, the server displays the message `Cache memory allocator exceeded minimum`.
- ♦ **Minimum File Cache Report Threshold** instructs the server to send an alert *before* the minimum number of file cache buffers is reached. Set this parameter to the number of buffers above the minimum at which you want the server to send an alert. For example, if the Minimum File Cache Buffers value is 20 and the Minimum File Cache Report Threshold value is 30, the server sends an alert when 50 file cache buffers are left.

eDirectory

If you suspect that your server is slow in accessing eDirectory™, see [Improving eDirectory Performance](#) in the *Novell eDirectory 8.7 Administration Guide*.

Tuning Directory Cache for the Traditional File System

The information for tuning directory cache applies only to NetWare servers using the traditional NetWare file system. Directory cache is not used by the Novell Storage Services (NSS) file system. You should only tune the directory cache if you are using the traditional file system.

As directory entries are read and operated upon by a user, NetWare caches the entries to make repeated use of an entry more efficient. In a default configuration, NetWare allocates 500 cache buffers of 4 KB each. Each block read into a cache buffer contains 32 entries.

The default number of buffers is appropriate for only a small number of users. The tuning strategies in this section can help you tune the directory cache to meet your network's requirements.

Three Directory Cache Tuning Strategies

Sizing the directory cache depends largely on the characteristics of the workload the server supports. The key is the frequency and breadth of directory searches, file opens, closes, and creations.

- ♦ A low-use scenario involves any number of users in which a small number of directories are shared or in which each user's activity is limited to a small region of the directory, such as a home directory.
- ♦ A high-use scenario involves any number of users, but user activity spans a very large number of directories and files. An extreme case might be a document-based system in which document searches routinely traverse large portions of a very large directory.

Strategy 1: Tuning for Low Use

To handle low memory use, you probably won't need to allocate any more cache than the NetWare default. By default, NetWare allocates 500 buffers immediately upon request, followed by a maximum allocation of up to 2000 buffers (about 2 MB). This is sufficient for the majority of low-use scenarios.

Strategy 2: Tuning for High Use

To tune for very high use, change the following set parameters to the values shown using NetWare Remote Manager, Monitor, or `set` commands at the System Console prompt.

```
minimum directory cache buffers = 2000
```

maximum directory cache buffers = 4000

In NetWare Remote Manager:

- 1 Click Set Parameters in the navigation frame > click Traditional File System, then select the parameter value link that you want to change.
- 2 Enter the new value, then click OK

In Monitor:

- 1 Click Available Options > Set Parameters > Traditional File System, then select the parameter you want to change.
- 2 Type the new value, then press Enter.

To specify a maximum number of concurrent writes of directory cache buffers, in NetWare Remote Manager, click Set Parameters > Common File System. The range is from 5-500.

Strategy 3: Tuning for Name Spaces

When a name space is installed on a traditional NetWare volume, the volume's Directory Entry Table (DET) must include an additional directory entry for each file. So, for example, if a volume supports DOS, NFS, and LONG name spaces, there are three directory entries in the DET for each file instead of just one.

In such a case, a directory cache buffer with 32 directory entries no longer represents 32 files, it represents only 10 files, because each file now requires three directory entries.

This means the efficiency of your directory cache is decreased by a factor equal to the number of name spaces loaded. Therefore, you might need to add more memory if you load multiple name spaces. By carefully tuning the directory cache, you can maintain high performance even under these conditions.

Before adding name spaces, allow your server to operate in its production environment for several weeks. This lets NetWare allocate the appropriate number of directory cache buffers for the native DOS environment.

After this settling-in period, check the Directory Cache Buffer value in the Monitor General Information window. This is the highest number of directory cache buffers allocated since the server was last started.

Use the following formulas to determine the optimum minimum and maximum number of directory cache buffers.

(# of names spaces, including DOS) x highest number of Directory Cache buffers since server was last started = minimum cache buffers
(minimum cache buffers) + 100 = maximum cache buffers

For example, if your server's highest number of Directory Cache buffers since the server was last started is 500, and you want to load one additional name space, you would calculate the minimum and maximum values of directory cache buffers as follows:

2 x 500 = 1000
2000 + 100 = 2100

Before loading the additional name space, change the following set parameter values using NetWare Remote Manager, Monitor, or **set** commands at the System Console prompt:

minimum directory cache buffers
maximum directory cache buffers

These settings increase the likelihood that repeatedly used directory cache buffers will remain in cache and that those buffers will remain in cache longer, providing the best possible read response times.

Monitoring the New Settings

After you tune the directory cache, let the server operate for a few weeks to settle in. Then check to see whether the server allocates the additional directory cache buffers that the new settings allow. If the allocated buffers do not increase to near the level allowed, then you know that your users' directory access patterns don't require the additional cache. On the other hand, if your server uses all the cache you made available, your user community's directory access patterns might be more intensive than you anticipated.

Using Virtual Memory

NetWare provides a virtual memory system that moves data out of memory and into a swap file on disk if the data isn't used frequently. Thus, the virtual memory system ensures that RAM is used more efficiently but with slower overall performance. It lessens the likelihood that low memory conditions will cause a problem with the server. For general information about virtual memory, see [“Virtual Memory” on page 14](#).

NetWare Remote Manager has several pages that can assist you in monitoring and managing the use of the virtual memory system in your server.

To	Do the following in NetWare Remote Manager
View the overall performance of the virtual memory system	<ol style="list-style-type: none">1. Click Health Monitor in the navigation frame.2. Click Virtual Memory Performance in the Server Health table.3. View the Virtual Memory Health Information page.
View specific performance of the virtual memory system	<ol style="list-style-type: none">1. Click View Memory Config in the navigation frame.2. Click Used by Virtual Memory System on the View Memory Config page.3. Click To Virtual Memory Cache Pool to view the statistics for each cache pool type. For information on each type of statistic listed, see the online help.4. Click To Virtual Memory Statistics to view the following statistics: Current Available Memory, Free Desired Pages, Free Lots Pages, Free Minimum Pages, Memory Available State, Need Free Clean Pages, Need Free Pages, and Upper Page I/O Threshold. For information on each type of statistic listed, see the online help.
Change set parameters associated with the virtual memory system	<ol style="list-style-type: none">1. Click Set Parameters in the navigation frame.2. Click Memory in the Category Name table on the Set Parameters page.3. Click the value link for each specific virtual memory parameter you want to change.4. Enter the new value in the Current Value field, then click OK.

One of the most important things you can do to allocate memory is to set the VM Cache Pool Percentage correctly. If you are running on a system with many applications running protected or many Java applications, set the percentage high. If you are not running these two conditions, the VM Cache Pool Percentage doesn't matter.

You can manage the swapping of data by controlling the swap files where the data is stored on disk. To create, delete, and manage swap files, access [NetWare Remote Manager](#) and complete the tasks in the following table:

To	Do the following in NetWare Remote Manager
Learn about the <code>swap</code> command	Click Console Commands in the navigation frame, then click SWAP Info on the Console Commands page.
Display the Swap File Information page	Click View Memory Config in the navigation frame > click Used by Virtual Memory System > click To Swap File Information.
Create a swap file on a designated volume	<ol style="list-style-type: none"> 1. Click View Memory Config in the navigation frame > click Used by Virtual Memory System > click To Swap File Information. 2. Click the value link in the Swapping Enabled column for the volume you want to enable. 3. (Optional) Enter the values you want in the Minimum Reserved, Maximum Used for Swapping, and Minimum Volume Free Space fields. These parameters specify minimum and maximum size of the swap file and the minimum free space that must be left on the volume. 4. On the Enable Swapping page, click Enable. 5. If the file is not on volume sys:, add the commands to create swap files to the autoexec.ncf file so the files are created each time the server is started. <p>The command syntax is <code>swap add volume_name [parameter=value...]</code></p> <p>Optional parameters are <code>min</code>, <code>max</code>, and <code>min free</code>.</p> <p>For example:</p> <pre>swap add vol1 min=3, max=6, min free=2000</pre>
Delete a swap file from a designated volume	<ol style="list-style-type: none"> 1. Click View Memory Config in the navigation frame > click Used by Virtual Memory System > click To Swap File Information. 2. Click the value link in the Swapping Enabled column for the volume you want to delete the Swap file from. 3. On the Disable Swapping page, click Disable. <p>If you are using protected address spaces, the Java Virtual Machine, or any other application that uses virtual memory, be sure to keep at least one swap file.</p> <p>If the swap file is being used when it is deleted, then the swapped data is moved to another swap file. If there is no other swap file and there is not enough physical memory to move all the swapped data, an error message is displayed and the file is not deleted.</p> <p>By default, a swap file is created on the volume sys:. If you do not want a swap file on volume sys:, place the <code>swap del sys</code> command in the startup.ncf file.</p>

To	Do the following in NetWare Remote Manager
Change the parameter values for a swap file on a designated volume	<ol style="list-style-type: none"> 1. Click View Memory Config in the navigation frame > click Used by Virtual Memory System > click To Swap File Information. 2. Click the link for the value you want to change. 3. On the Change Swap Parameters page, enter the new values you want, then click Change.

Using Swap Files

You can create one swap file per volume. The swap file for volume sys: is created by default; you can delete it if necessary or move it to another volume.

Data moved to disk by virtual memory is stored in any available swap file; it does not matter which volume the swap file is on. Generally, you should place swap files on the fastest volume, the one with the most available space, or the one that is used the least.

You can add a swap file to a volume that is not mounted. The file is created when you mount the volume.

When you dismount a volume, the swap file is deleted. To keep a swap file on that volume, you must create the swap file again when the volume is mounted. The exception is the swap file for volume sys:, which is created by default. To recreate swap files on volumes other than sys:, add the appropriate commands to the autoexec.ncf file, so the files are created each time the server is started.

If you do not add the appropriate commands to autoexec.ncf, all swap files on volumes other than sys: are permanently deleted.

If you want to delete the swap file on sys: each time the server is restarted, add the appropriate command to autoexec.ncf.

Swap files are dynamic; they expand and contract as data is swapped in and out of memory.

You can specify the minimum, maximum, and minimum free settings for swap files on a designated volume. If you do not specify swap file settings, the default minimum swap file size is 2 MB, the default maximum swap file size is the free volume space, and the default minimum free space on the volume is 5 MB.

Responding to Disk Thrashing

The real value of virtual memory is in using a sufficient supply of memory more efficiently, thereby improving server performance.

If the overall supply of memory is running low, then swapping will occur more often when virtual memory is in use. If memory is extremely low, the system might spend a significant amount of its time swapping memory in and out of disk and have little time to accomplish useful work. This is called *disk thrashing*.

In extremely low memory conditions, NetWare moves all the memory from a protected address space into the swap file, temporarily stopping the modules within the space. After a period of time, NetWare moves the memory back into the protected address space and shuts down another space, moving its memory to the swap file. Without virtual memory, these extremely low memory conditions would cause processes to fail. With virtual memory, the server keeps running, although very slowly.

If the Average Clean Free Available state or the Average Page I/O state is Suspect or Bad on the Virtual Memory Health Information page in NetWare Remote Manager, your server is experiencing disk thrashing.

The best way to prevent disk thrashing is to control how much memory is used by the virtual memory system. Set the VM Cache Pool Percentage parameter to a percentage between 0 - 90.

If you set it to more than the virtual memory system needs, it is harmless because the virtual memory system will not take it. But if you set it to less than the virtual memory system needs, you will create disk thrashing.

To set the VM Cache Pool Percentage, do the following:

- 1** In NetWare Remote Manager, click Set Parameters.
- 2** Under Set Parameter Control, click View Hidden Set Parameters to On.
- 3** Under Category Name, click Memory > VM Cache Pool Percentage.
- 4** In the Current Value field, enter the new value, then click OK.

If your server is experiencing disk thrashing, you can also use the following two hidden set parameters (Memory category) to manage the balance between the virtual memory cache pool and the file system cache pool.

- ◆ VM Cache Pool Free Desired Pages, range 0 to 2147483647, default 256
This parameter specifies the number of pages that the virtual memory system cache needs to operate properly.
- ◆ VM Cache Pool Free Lots of Pages, range 0 to 2147483647, default 512
This parameter specifies the number of pages that the virtual memory system cache considers to be excessive. Having an available page count higher than this number might result in the pages being given to another cache pool.

You can specify the point at which the server sends an alert to the console because of excessive memory swapping by using the Average Page In Alert Threshold Set parameter (Memory category).

The value is the average number of pages swapped from disk to memory per second, calculated over the last five seconds. The default is 2000.

To change the values for this parameter using NetWare Remote Manager, do the following:

- 1** Click Set Parameters in the navigation frame, then click Memory in the Category Name table on the Set Parameters page.
- 2** Click the value link for the Average Page In Alert Threshold setting.
- 3** Enter the new value in the Current Value field, then click OK.

When disk thrashing occurs, the complete solution is to add more RAM. Virtual memory cannot compensate for an overall lack of server memory, although it can prevent processes from failing and allow a server to continue to function.

Optimizing Garbage Collection

The NetWare server provides a **garbage collection** or memory recovery process that periodically collects allocated memory freed by NLM programs and returns it to cache. Only the traditional file

system uses garbage collection. For general information on garbage collection, see “[Garbage Collection](#)” on page 12.

Garbage collection responds to pressure from the virtual memory system. (For an explanation of virtual memory, see “[Virtual Memory](#)” on page 14.)

If pressure on virtual memory is high, then garbage collection happens immediately.

If pressure on virtual memory is low, then garbage collection happens at the regularly scheduled garbage collection interval. This interval is controlled by the Garbage Collection Interval server parameter.

To view or change this interval in the NetWare Remote Manager, in the navigation frame click Set Parameters > Memory > Garbage Collection Interval.

Normally, garbage collection happens in the background. The only indicator that garbage collection has taken place is an increase in the amount of free memory displayed in Monitor after garbage collection has freed memory.

However, you can initiate garbage collection manually for individual modules using NetWare Remote Manager or Monitor.

In NetWare Remote Manager:

- 1** Click List Modules.
- 2** In the field, type the name of module that you want to garbage collect memory from, then click Load Module.
- 3** Click the garbage collection link for that module.

In Monitor:

- 1** Click Available Options > Loaded Modules.
- 2** Select the module that you want to garbage collect memory from.
- 3** Press F4 to garbage collect memory for that module.

HINT: You can also garbage collect memory from protected address spaces. In Monitor, click Available Options > select Virtual Memory > click Address Spaces > select the desired address space > press F4.

