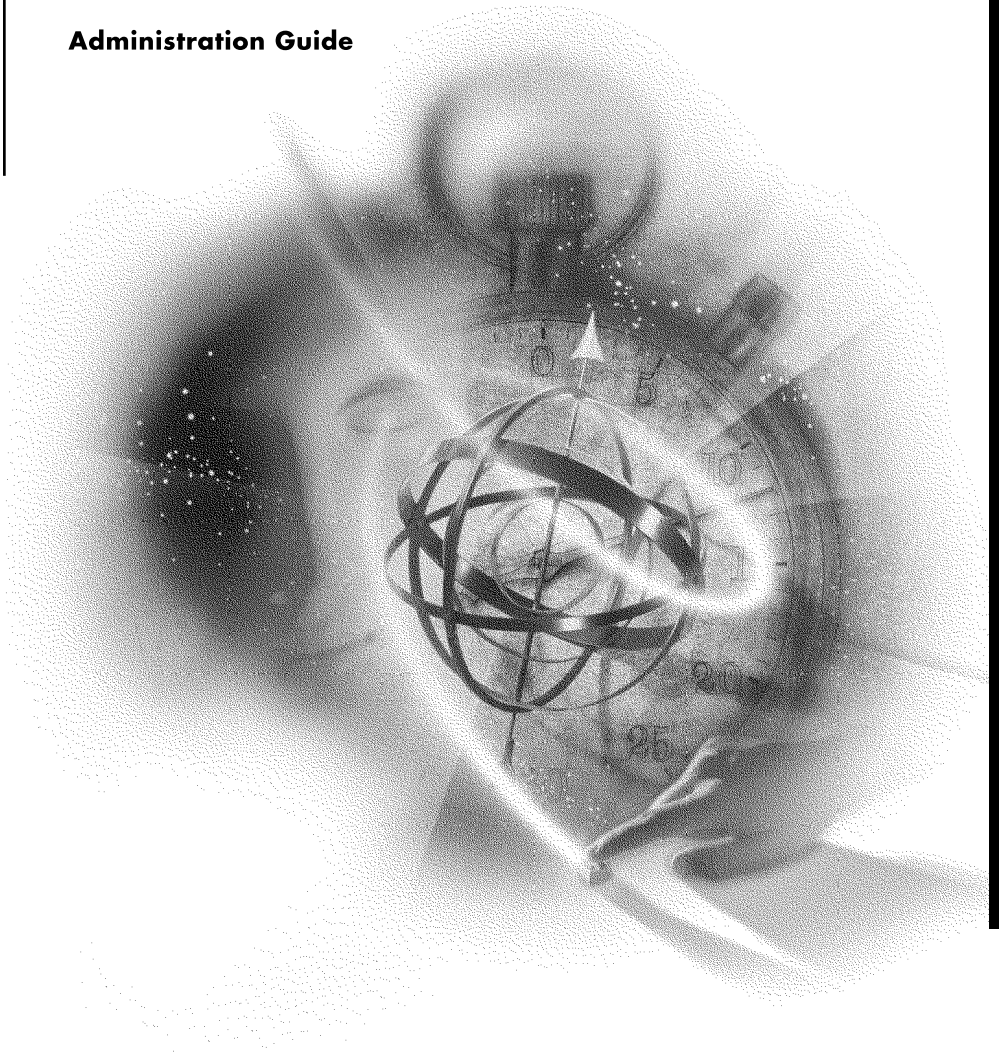
NetWare Server Memory

Administration Guide



NetWare® 5.1
NETWORKING SOFTWARE

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.

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

U.S. Patent Nos. 4,555,775; 5,157,663; 5,349,642; 5,455,932; 5,553,139; 5,553,143; 5,594,863; 5,608,903; 5,633,931; 5,652,854; 5,671,414; 5,677,851; 5,692,129; 5,758,069; 5,758,344; 5,761,499; 5,781,724; 5,781,733; 5,784,560; 5,787,439; 5,818,936; 5,828,882; 5,832,275; 5,832,483; 5,832,487; 5,859,978; 5,870,739; 5,873,079; 5,878,415; 5,884,304; 5,893,118; 5,903,650; 5,905,860; 5,913,025; 5,915,253; 5,925,108; 5,933,503; 5,933,826; 5,946,467; 5,956,718; 5,974,474. U.S. and Foreign Patents Pending.

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

www.novell.com

NetWare Server Memory Administration Guide
January 2000
104-001240-001

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

Novell Trademarks

For a list of Novell trademarks, see the final appendix of this book.

Third-Party Trademarks

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

Contents

Server Memory

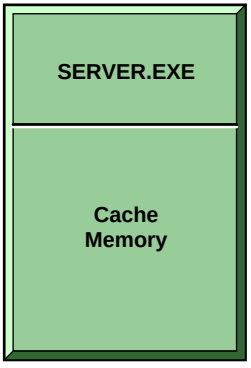
NetWare provides increased memory efficiency with three important features: logical memory addressing, virtual memory, and protected memory. For a general introduction to these and other aspects of NetWare's memory architecture, see "Understanding" on page 9.

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

- The MONITOR utility provides statistics that indicate how efficiently memory is being used by the server and individual modules. For information about memory statistics and general memory management, see "Managing" on page 31.
- Settable parameters and console commands let you tune the memory system to achieve optimal performance. For information on tuning memory, see "Optimizing" on page 17.

1 Understanding

NetWare requires a minimum of 64 MB of RAM and can access up to 4 GB of RAM. After the server.exe file is loaded, all remaining memory is assigned to cache memory, which is available for NLM programs and other processes to use. When data is stored in cache, it is stored in 4 KB blocks called cache buffers.



For more information about NetWare's use of cache memory, see "Introduction to NetWare Memory" on page 10.

NetWare provides a logical memory addressing scheme that minimizes memory fragmentation. See "Logical Memory Addressing" on page 12.

Troublesome or untried applications can now be loaded into protected memory to isolate them from the server kernel. Applications running in protected address spaces cannot abend the server. For information about protected memory, see "Protected Address Spaces" on page 13.

NetWare also provides a virtual memory system that swaps little-used data to disk, thus freeing memory for more frequently used data. Applications that run in protected address spaces use virtual memory, as does Java. For general information about the virtual memory system, see “Virtual Memory” on page 15.

Introduction to NetWare Memory

Recently used data is kept in cache buffers in case it is needed again. Caching data speeds processing because data can be written to cache or read from cache hundreds of times faster than it can be read from or written to disk.

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

Storing volumes’ FAT and suballocation tables

Storing volumes’ directory entry tables

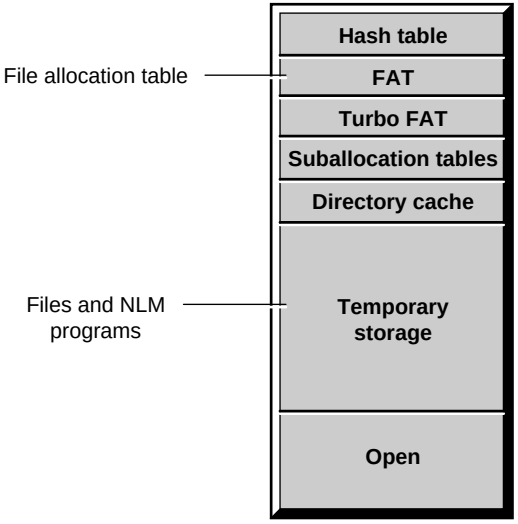
Storing data files for users

Storing NLM programs, such as LAN drivers, disk drivers, and management utilities

Building hash tables for directory names

Creating protected address spaces

Figure 1 Example of disk cache allocation



Memory from cache is allocated to a process when requested. When a process is through using the memory, it frees the memory and returns it to cache. NetWare also provides a garbage collection process to recover unused memory. The garbage collector runs in the background, but you can manually initiate the process if needed. See “Optimizing Garbage Collection” on page 28.

Directory Cache 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. 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 also 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 Server Memory” on page 19.

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. File cache stores recently used file data and also parts of the NDS® database.

File cache buffers are organized in a structure called a linked list. When the server receives a request for data that is on disk, it retrieves the data and stores it in a file cache buffer. The buffer is placed at the head of the linked list and is stamped with the current time. This buffer is called the most recently used (MRU) buffer. As new buffers are added to the head of the list, the older buffers move down the list. Whenever the server requires a buffer to store data, it retrieves the oldest buffer from the tail of the list, stores the new data in the buffer, and moves the buffer to the top of the list.

When the server receives a request for data that is already in a cache buffer, it retrieves the data from the buffer, stamps the buffer with the new time, and moves the buffer to the head of the list. This way, the most recently used buffers are found at or near the top of the list and the least recently used (LRU) buffers gather at the bottom.

In an ideal environment, there is enough memory so that all frequently used data can be retrieved from file cache rather than from disk. The MONITOR utility provides a disk cache utilization statistic, called the LRU Sitting Time, that helps you estimate whether your server has sufficient memory to service requests from cache. For information about using the LRU Sitting Time statistic, see “Tuning File Cache” on page 22.

Logical Memory Addressing

The physical memory of the server is divided into 4 KB blocks, called pages, as defined by Intel. When a process 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.

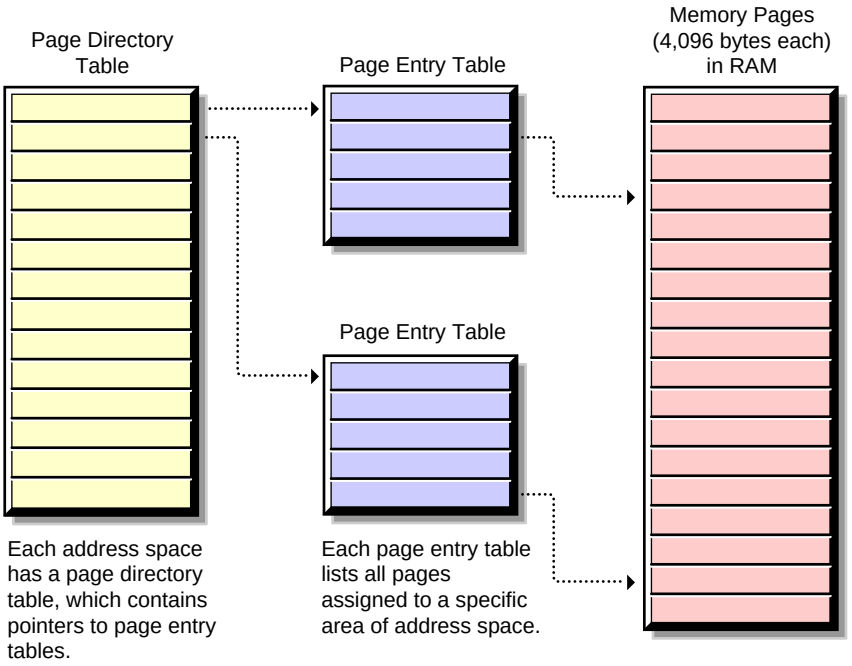
As contiguous blocks of memory pages are used up, small segments of memory are often left over. After awhile, there are many such isolated pieces of memory too small to be used. This situation is called memory fragmentation. When memory is fragmented, the server often can’t satisfy memory requests.

NetWare’s logical memory addressing scheme helps to prevent memory fragmentation The memory management subsystem provides a page directory table that lists logical memory addresses from 0 to 4 GB. The table lists addresses up to 4 GB even if the server contains only the minimum 64 MB of RAM.

When a process needs a contiguous block of memory, the operating system finds a set of contiguous addresses in the page directory table. Because the table has so many addresses (4 GB), the operating system can almost always find the contiguous memory addresses the processes need.

Other tables, called page entry tables, translate the logical addresses to physical addresses, which might not be contiguous at all, but which might be scattered throughout RAM. As long as the logical addresses are contiguous, it does not matter that the physical memory pages are not contiguous. Together, the page directory table and the page entry tables are called the page translation table.

Figure 2 Page translation table



Protected Address Spaces

Portions of cache memory can be set aside as protected address spaces (sometimes called user address spaces or ring 3). A protected address space is a portion of cache memory that has carefully controlled communication with the server operating system. All protected address spaces use virtual memory. For general information about virtual memory, see “Virtual Memory” on page 15.

You can use protected address spaces to run untried or troublesome applications. Because any 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 cannot run in a protected address space. The operating system address space (ring 0) is sometimes called the OS address space or the kernel address space.

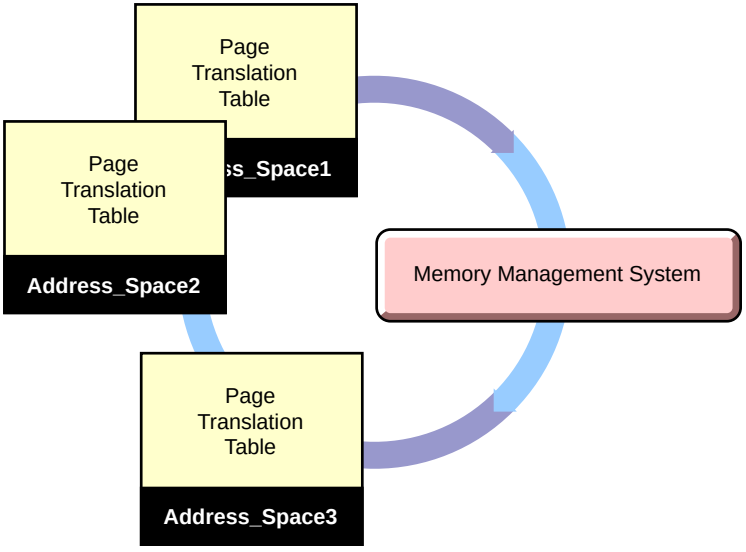
The syscalls.nlm, in conjunction with the memory protection subsystem, prevents modules in a protected address space from having direct access to anything outside the address space. In particular, syscalls.nlm and the memory protection subsystem serve as the interface between the protected address spaces and the server operating system. They prevent NLM programs in protected spaces from passing calls to the operating system address space that would corrupt or fault the core operating system.

You can load modules into a protected address space, unload modules from a space, delete a space, or kill a space. See “Loading Protected Address Spaces” on page 32 and “Unloading Protected Address Spaces” on page 35.

Protected Address Spaces Use Logical Memory Addressing

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

Only one translation table can be loaded into the memory management subsystem at a time. When there is more than one protected address space, the memory management subsystem loads one translation table for a specified time, then replaces it with another.



Replacing translation tables is called a context switch. Context switches are somewhat time consuming; do not create more protected address spaces than you need.

All protected address spaces use virtual memory. See “Virtual Memory” on page 15.

Modules That Cannot Run in Protected Address Spaces

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

- server.exe

  The server operating system cannot run in a protected address space because any software in a protected space is subject to being temporarily stopped and swapped to disk. (See “Virtual Memory” on page 15.)

- LAN and disk drivers

  Hardware and LAN drivers must run in the operating system address space because they use only physical memory addresses. Protected spaces use logical memory addressing.

- MONITOR

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

- The NSS file system

- Other modules or applications that aren’t designed to run in protected address spaces.

Virtual Memory

Virtual memory provides more efficient use of memory and lessens the likelihood that low memory conditions will cause an abend.

Available memory is assessed to see which data has been used less recently than the rest. How recently data has been used is determined by a bit in each field of the translation table that indicates whether the address translation has been used since the last time it was checked. Data that has not been used for some time can be moved from memory to disk, thus 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 disk, the available memory can be used for a larger amount of data than its actual physical capacity would allow.

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.

Server operating system modules do not use virtual memory, because they cannot be swapped to disk.

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

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 the Sys volume by default.

You can use the SWAP command to create a swap file, delete a swap file, change swap file parameters, and view information about the swap file. You can also view information about the swap file in MONITOR. See "Using Virtual Memory" on page 25.

Swapping the data between memory and disk requires additional system resources, but it increases the memory available for use. The availability of the additional memory can improve overall server performance.

2 Optimizing

To tune your server's memory, you must understand how to interpret key memory statistics. Interpreting key statistics allows you to

- Assess the amount of RAM in your server and determine how well it's being used.
- Monitor the memory use of individual modules.
- Tune directory and file cache.
- Know when to increase file cache buffers or add more RAM.

Assessing Server RAM

Insufficient physical memory (also known as RAM) in the server is a common cause of a slow network. Use this procedure to determine whether your server has sufficient physical memory installed.

You can first check the amount of server memory currently available for disk cache from MONITOR's Available Options > System Resources > Server Memory Statistics > Cache Buffer Memory.

1. In MONITOR's General Information window, locate Long Term Cache Hits.
2. Note the percentage of Long Term Cache Hits displayed.

   The percentage of Long Term Cache Hits should exceed 90 percent. If it falls below this value, the solution is to 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 treat symptoms only and are not long-term solutions.

3 To exit MONITOR, press Alt+F10.

For more information, see “Introduction to NetWare Memory” on page 10 and Reference > *Utilities Reference* > Utilities > MONITOR.

Determining Which Processes Use Too Much Memory

Use this procedure to determine which NLM programs are using excessive amounts of memory.

1 From MONITOR’s Available Options, select Loaded Modules.

A list of loaded modules appears.

2 Highlight a module 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 highlighted module.

Remember that the memory recovery operation also uses memory.

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 = 200
```

The default number of buffers is 200. 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. See Matching Performance Capabilities for a discussion of network bottlenecks.

For more information about the parameter, see Reference > *Utilities Reference* > Utilities > SET > Memory Parameters. To set the parameter, load MONITOR, select > Server Parameters > Memory.

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. See Reference > *Utilities Reference* > Utilities > REGISTER MEMORY.

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 NetWare's automatic memory registration will work.

For more information about memory management and registering memory, see the Novell knowledgebase at http://support.novell.com.

Tuning Server Memory

Although NetWare is a self-tuning operating system, 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. Once 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 procedures.

The procedures for tuning your server's memory apply only to the traditional NetWare file system. In contrast, the Novell Storage Services (NSS) file system provides improved resource use, requiring less memory to mount multiple volumes. For information about NSS, see Contents > Novell Storage Services > Understanding > The Benefits of NSS.

To tune the server's directory cache, see "Tuning Directory Cache" on page 20. To tune the server's file cache, see "Tuning File Cache" on page 22. The information in these two sections is from Optimizing IntranetWare Server Memory, in the March 1997 *AppNotes*.

Tuning Directory Cache

The following information applies only to NetWare running the traditional NetWare file system. Directory cache is not used by the NSS 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 20 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 will 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 NetWare's default. By default, NetWare allocates 20 buffers immediately upon request, followed by a maximum allocation of up to 500 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, use the SET command at the server console prompt or the MONITOR utility (MONITOR > Available Options > Server Parameters > Directory Caching) to change the directory cache buffer server parameters as follows:

```
MINIMUM DIRECTORY CACHE BUFFERS = 2000
MAXIMUM DIRECTORY CACHE BUFFERS = 4000
```

Strategy 3: Tuning for Name Spaces

When a name space is installed on a traditional NetWare volume, the volume's 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 only represents 10 files, since 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, look at the MONITOR General Information window to see the value for Directory Cache Buffers. This value is a high watermark: it 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 watermark = minimum
  cache buffers
(minimum cache buffers) + 100 = maximum cache buffers
```

For example, if your server's directory cache buffer watermark is 250, 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 250 = 500
500 + 100 = 600
```

Before loading the additional name space, use the SET command or MONITOR to change the minimum and maximum directory cache buffer parameters (MONITOR > Available Options > Server Parameters > Directory Caching).

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 may be more intensive than you anticipated.

Tuning File Cache

File cache not only speeds access to file data, it is used to cache portions of the NDS® database. If you want to tune your NetWare server in general, or NDS in particular, tune the file cache.

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 tail of the list. The length of time the oldest buffer has been in the list is shown as the LRU Sitting Time statistic in MONITOR's Cache Utilization Statistics screen (MONITOR > Available Options > Disk Cache Utilization.)

In tuning file cache, 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 statistic (MONITOR > Available Options > Disk Cache Utilization), 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 at which 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 will 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 new data is not being 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.

Calculating Cache Memory Needs

Follow these steps to determine how much file cache your server needs.

1. Track server resource utilization statistics found in MONITOR. 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. This average is the low watermark.

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

   For example, if your server's original low watermark was 7 minutes, you will need to add enough memory to increase the LRU Sitting Time to an average 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 low watermark 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. Use the following server parameters, found in MONITOR > Available Options > Server Parameters > File Caching:

- 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 cache buffer limit. Increase the value of the parameter if necessary.
- 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.

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. It lessens the likelihood that low memory conditions will cause an abend.

For general information about virtual memory, see "Virtual Memory" on page 15.

You can manage the swapping of data by controlling the swap files where the data is stored on disk. Use the following commands to create, delete, and manage swap files.

To...	At the server console prompt, enter...
Learn about the SWAP command	HELP SWAP
Display a Virtual Memory Information Screen containing information about swap files	SWAP Alternatively, see MONITOR > Available Options > Virtual Memory > Swap Files.
Create a swap file on a designated volume.	SWAP ADD *volume_name* [*parameter= value...*] Optional parameters are MIN=, MAX=, and MIN FREE=. These parameters specify minimum and maximum size of the swap file and the minimum free space that must be left on the volume. Values are in millions of bytes. If parameters are not included, default values are used: MIN = 2 MAX = Free volume space MIN FREE = 5

Delete a swap file from a designated volume	SWAP DEL *volume_name* 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. 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, an error message is displayed and the file is not deleted. By default, a swap file is created on the Sys volume. If you do not want a swap file on Sys, place the SWAP DEL command in the startup.ncf file before the command to mount volumes.
Change the parameter values for a swap file on a designated volume	SWAP PARAMETER *volume_name parameter=value...* Parameters are MIN=, MAX=, and MIN FREE=. These parameters specify the minimum and maximum size of the swap file and the minimum free space that must be left on the volume. Values are in millions of bytes.
Specify the point at which the server sends an alert to the console because of excessive memory swapping.	SET AVERAGE PAGE IN ALERT THRESHOLD = *value* 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. If the average page-in rate is high, it means a large amount of data is being swapped to and from disk, indicating that the server might be running low on memory. You can also set the parameter from MONITOR > Available Options > Server Parameters > Memory.

Using Swap Files

- You can create one swap file per volume. The swap file for the Sys volume is created by default; you can delete it if necessary.
- Data moved to disk by virtual memory will be stored in any available swap file; it does not matter which volume the swap file is on. Generally, you will place swap files on the fastest volume or the one with the most available space.
- You can add a swap file to a volume even if the volume is not mounted. Once you've added the swap file, the file will be 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. The exception is the Sys volume swap file, which is created by default. For convenience, place the commands to create swap files into the autoexec.ncf file, so the files will be created each time the server is started.
- Swap files are dynamic; they change size as data is swapped in and out of memory.

Responding to Disk Thrashing

If the overall supply of memory is running low, then swapping will occur more often. If memory is extremely low, the system may spend all its time swapping memory in and out of disk and have no time to accomplish useful work. This is called disk thrashing.

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

When disk thrashing occurs, the 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.

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

Optimizing Disk Cache Utilization

Because it is much faster to get data from memory than to get it from disk, caching frequently requested data will increase 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.

1. From MONITOR's Available Options, select Disk Cache Utilization.

   The Cache Utilization Statistics appear in the upper window.

2. Locate the Short Term Cache Hits field and note the value.

   Short-term hits occurred in the last second. This value should be over 98%.

3. Locate the Long Term Cache Hits field and note the value.

   Long-term hits are a cumulative percentage since the server was last started. This value should be over 90%. (This field also appears on the MONITOR General Information window.)

4. If the values of these two fields are not acceptable, add more RAM.

   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 treat symptoms only and are not a long-term solutions.

Optimizing Garbage Collection

The NetWare server provides a garbage collection or memory recovery process that periodically collects alloc memory freed by NLM programs and returns it to cache.

Garbage collection is triggered by pressure from the virtual memory system. (For an explanation of virtual memory, see "Virtual Memory" on page 15.) The virtual memory system is checked every 10 seconds.

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, access the parameter in the MONITOR utility (MONITOR > Available Options > Server Parameters > Memory).

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. However, you can initiate garbage collection manually for individual modules. Follow the steps below.

1. From the MONITOR Available Options menu, select Loaded Modules.
2. Highlight the module from which you want to free alloc memory.
3. Press F4 to free memory for that module.

   NOTE: You can also free memory for protected address spaces. In MONITOR's Available Options, select Virtual Memory > Address Spaces. Then highlight the desired address space and press F4.

3 Managing

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

Only applications and files stored in the traditional NetWare file system can be loaded into protected memory. The Novell Storage Services (NSS) file system does not use protected memory; it runs in the kernel address space.

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 13. For general information about virtual memory, see "Virtual Memory" on page 15.

Loading Protected Address Spaces

You can create a protected address space when you load one or more modules. Use the following commands at the server console prompt.

Command	Description	Example
[LOAD] PROTECTED *module_name*	Loads one module into a new protected address space and names the space ADDRESS_SPACE*n*.	[LOAD] PROTECTED DATABASE.NLM Creates a new address space called ADDRESS_SPACE*n*, where *n* is a number, and loads database.nlm into it.
[LOAD] RESTART *module_name*	Loads one module into a new protected space with restart functionality and names the space ADDRESS_SPACE*n*. 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 parameter. (MONITOR > Available Options > Server Parameters > Memory.) See Memory Parameters for more information.	RESTART DATABASE.NLM Creates a new address space called ADDRESS_SPACE*n*, where *n* is a number, and loads database.nlm into it. If the address space abends, the system will shut down the space, restart the space automatically, and reload database.nlm into it.

[LOAD] ADDRESS SPACE = *space_name module_name*	Loads one module into a protected space with a user-defined name. Use this command when you want to create your own name for the space. You can 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.	[LOAD] ADDRESS_SPACE = DB_SPACE DATABASE.NLM Creates a new address space called DB_SPACE and loads database.nlm into it. [LOAD] ADDRESS_SPACE = DB_SPACE RESTART TEST.NLM Loads test.nlm into the DB_SPACE address space and also adds restart functionality to the address space.
PROTECT *filename*	Use this command when you normally use a .ncf file to load multiple modules at once. 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.	PROTECT LOAD_DB Creates an address space called LOAD_DB and executes load_db.ncf to load modules into the protected space.
PROTECTION RESTART *space_name*	Adds restart functionality to an existing address space. If the protected space abends, the system closes the space, cleans up its resources, restarts the space, and reloads modules into it. To prevent the server from restarting a memory space that continues to fault, use the Memory Protection No Restart Interval parameter. (MONITOR > Available Options > Server Parameters > Memory.) See Memory Parameters for more information.	PROTECTION RESTART ADDRESS_SPACE3 Adds restart functionality to ADDRESS_SPACE3.

PROTECTION NO RESTART *space_name*	Removes restart functionality from an existing address space.	PROTECTION NO RESTART ADDRESS_SPACE3 Removes restart functionality from ADDRESS_SPACE3.

Using Protected Address Spaces

- Because modules loaded into a protected address space have controlled communication with the operating system, all modules that must communicate with each other, such as the modules of an e-mail application or a database, should be loaded into the same protected address space. Use the LOAD ADDRESS SPACE = command to load multiple modules into the same space.
- If you load an 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, if you load them into different protected 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 space.
- If you want the server to clean up address spaces that abend, set the Memory Protection Fault Cleanup parameter to On. If an address space abends, the server removes the space and its modules and returns the resources to the system. Access the parameter from MONITOR > Available Options > Server Parameters > Memory.

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

Unloading Protected Address Spaces

To unload a module from a protected space or to unload the space itself, use the following commands at the server console prompt.

Command	Description	Example
UNLOAD ADDRESS SPACE = *space_name module_name*	Unloads the specified module from the specified address space. If the module is loaded into more than one address space, this command removes the module only from the designated address space.	UNLOAD ADDRESS SPACE = ADDRESS_SPACE1 DATABASE.NLM Unloads database.nlm from ADDRESS_SPACE1, but does not remove the address space.
UNLOAD ADDRESS SPACE = *space_name*	Unloads all modules from the designated address space and removes the space.	UNLOAD ADDRESS SPACE = ADDRESS_SPACE1 Unloads all the modules from ADDRESS_SPACE1, closes ADDRESS_SPACE1, and returns its resources to the system.
UNLOAD KILL ADDRESS SPACE = *space_name*	Removes the address space but without unloading the modules first. Use this command only if you know the space can't be unloaded with the UNLOAD ADDRESS SPACE command.	UNLOAD KILL ADDRESS SPACE=ADDRESS_SPACE1 Removes ADDRESS_SPACE1 without unloading modules . Returns the resources of ADDRESS_SPACE1 to the system.

- 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 parameter. The default is 30 seconds. You can access the parameter in the MONITOR utility (MONITOR > Available Options > Server Parameters > Error Handling).

- 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 console 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 and Virtual Memory

To see which modules are loaded in which address spaces, execute either the MODULES or the PROTECTION command at the server console prompt, as follows:

1. To display a list of loaded modules with their address space names, enter the following at the server console prompt:

   MODULES

   Each NLM name 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 the following at the server console prompt:

   PROTECTION

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

Additional information is accessible in MONITOR. To view information about virtual memory, as well as statistics for address spaces and swap files, follow these steps.

1 From MONITOR's Available Options, select Virtual Memory.

MONITOR displays the Virtual Memory Options menu and an information window. Statistics in the window show the number of pages swapped to and from disk and the number of swap file pages available.

1a Press Tab to expand and activate the information window.

1b Press F1 to see a description of each field. Press Esc to exit the window.

2 Select Address Spaces from the Virtual Memory Options menu.

MONITOR displays both a list of address spaces and an information window.

2a Use the arrow keys to highlight an address space in the list

The information window displays data about the highlighted address space.

2b Press Tab to activate and expand the information window.

2c Press F1 to see a description of each field. Press Esc twice to exit the window.

2d Highlight a specific address space and press Enter to see a list of all modules loaded into the address space.

2e Use the arrow keys to highlight a module in the list.

The information window displays data about the highlighted module. Press Tab to activate and expand the window. Press F1 to see a description of each field.

2f Press Esc until the Virtual Memory Options menu is displayed.

3 Select Swap Files from the Virtual Memory Options menu.

MONITOR displays both a list of swap files and an information window.

3a Use the arrow keys to highlight a swap file.

The information window displays data about the highlighted swap file.

3b Press Tab to expand and activate the information window.

3c Press F1 to see a description of each field.

3d Press Esc until the MONITOR Available Options menu is displayed.

A Novell Trademarks

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

Advanced NetWare is a trademark of Novell, Inc.

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

AppNotes is a registered service mark of Novell, Inc. in the United States and other countries.

AppNotes is a registered service mark of Novell, Inc. in the United States and other countries.

AppTester is a registered service mark of Novell, Inc. in the United States and other countries.

BrainShare is a registered service mark of Novell, Inc. in the United States and other countries.

C-Worthy is a trademark of Novell, Inc.

C3PO is a trademark of Novell, Inc.

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

Certified NetWare Administrator in Japanese and CNA-J are service marks of Novell, Inc.

Certified NetWare Engineer in Japanese and CNE-J are service marks of Novell, Inc.

Certified NetWare Instructor in Japanese and CNI-J are service marks of Novell, Inc.

Certified Novell Administrator and CNA are service marks of Novell, Inc.

Certified Novell Engineer is a trademark and CNE is a registered service mark of Novell, Inc. in the United States and other countries.

Certified Novell Salesperson is a trademark of Novell, Inc.

Client 32 is a trademark of Novell, Inc.

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

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

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

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

Custom 3rd-Party Object and C3PO are trademarks of Novell, Inc.

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

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

ElectroText is a trademark of Novell, Inc.

Enterprise Certified Novell Engineer and ECNE are service marks of Novell, Inc.

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

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

EXOS is a trademark of Novell, Inc.

Global MHS is a trademark of Novell, Inc.

Global Network Operations Center and GNOC are service marks of Novell, Inc.

Graphics Environment Manager and GEM are registered trademarks of Novell, Inc. in the United States and other countries.

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

GroupWise XTD is a trademark of Novell, Inc.

Hardware Specific Module is a trademark of Novell, Inc.

Hot Fix is a trademark of Novell, Inc.

InForms is a trademark of Novell, Inc.

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

Internetwork Packet Exchange and IPX are trademarks of Novell, Inc.

IPX/SPX is a trademark of Novell, Inc.

IPXODI is a trademark of Novell, Inc.

IPXWAN is a trademark of Novell, Inc.

LAN WorkGroup is a trademark of Novell, Inc.

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

LAN WorkShop is a trademark of Novell, Inc.

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

LANalyzer Agent is a trademark of Novell, Inc.

Link Support Layer and LSL are trademarks of Novell, Inc.

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

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

Media Support Module and MSM are trademarks of Novell, Inc.

Mirrored Server Link and MSL are trademarks of Novell, Inc.

Mobile IPX is a trademark of Novell, Inc.

Multiple Link Interface and MLI are trademarks of Novell, Inc.

Multiple Link Interface Driver and MLID are trademarks of Novell, Inc.

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

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

Natural Language Interface for Help is a trademark of Novell, Inc.

NDS Manager is a trademark of Novell, Inc.

NE/2 is a trademark of Novell, Inc.

NE/2-32 is a trademark of Novell, Inc.

NE/2T is a trademark of Novell, Inc.

NE1000 is a trademark of Novell, Inc.

NE1500T is a trademark of Novell, Inc.

NE2000 is a trademark of Novell, Inc.

NE2000T is a trademark of Novell, Inc.

NE2100 is a trademark of Novell, Inc.

NE3200 is a trademark of Novell, Inc.

NE32HUB is a trademark of Novell, Inc.

NEST Autoroute is a trademark of Novell, Inc.

NetExplorer is a trademark of Novell, Inc.

NetWare Link/ATM is a trademark of Novell, Inc.

NetWare Link/Frame Relay is a trademark of Novell, Inc.

NetWare Link/PPP is a trademark of Novell, Inc.

NetWare Link/X.25 is a trademark of Novell, Inc.

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

NetWare LU6.2 is trademark of Novell, Inc.

NetWare Management Agent is a trademark of Novell, Inc.

NetWare Management System and NMS are trademarks of Novell, Inc.

NetWare Message Handling Service and NetWare MHS are trademarks of Novell, Inc.

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

NetWare Mirrored Server Link and NMSL are trademarks of Novell, Inc.

NetWare Mobile is a trademark of Novell, Inc.

NetWare Mobile IPX is a trademark of Novell, Inc.

NetWare MultiProtocol Router and NetWare MPR are trademarks of Novell, Inc.

NetWare MultiProtocol Router Plus is a trademark of Novell, Inc.

NetWare Name Service is trademark of Novell, Inc.

NetWare Navigator is a trademark of Novell, Inc.

NetWare Peripheral Architecture is a trademark of Novell, Inc.

NetWare Print Server is a trademark of Novell, Inc.

NetWare Ready is a trademark of Novell, Inc.

NetWare Requester is a trademark of Novell, Inc.

NetWare Runtime is a trademark of Novell, Inc.

NetWare RX-Net is a trademark of Novell, Inc.

NetWare SFT is a trademark of Novell, Inc.

NetWare SFT III is a trademark of Novell, Inc.

NetWare SNA Gateway is a trademark of Novell, Inc.

NetWare SNA Links is a trademark of Novell, Inc.

NetWare SQL is a trademark of Novell, Inc.

NetWare Storage Management Services and NetWare SMS are trademarks of Novell, Inc.

NetWare Telephony Services is a trademark of Novell, Inc.

NetWare Tools is a trademark of Novell, Inc.

NetWare UAM is a trademark of Novell, Inc.

NetWare WAN Links is a trademark of Novell, Inc.

Novell Gold Authorized Reseller is a service mark of Novell, Inc.

Novell Gold Partner is a service mark of Novell, Inc.

Novell Labs is a trademark of Novell, Inc.

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

Novell NE/2 is a trademark of Novell, Inc.

Novell NE/2-32 is a trademark of Novell, Inc.

Novell NE3200 is a trademark of Novell, Inc.

Novell Network Registry is a service mark of Novell, Inc.

Novell Platinum Partner is a service mark of Novell, Inc.

Novell Press is a trademark of Novell, Inc.

Novell Press Logo (teeth logo) is a registered trademark of Novell, Inc. in the United States and other countries.

Novell Replication Services is a trademark of Novell, Inc.

Novell Research Reports is a trademark of Novell, Inc.

Novell RX-Net/2 is a trademark of Novell, Inc.

Novell Service Partner is a trademark of Novell, Inc.

Novell Storage Services is a trademark of Novell, Inc.

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

Novell Technical Services and NTS are service marks of Novell, Inc.

Novell Technology Institute and NTI are registered service marks of Novell, Inc. in the United States and other countries.

Novell Virtual Terminal and NVT are trademarks of Novell, Inc.

Novell Web Server is a trademark of Novell, Inc.

Novell World Wide is a trademark of Novell, Inc.

NSE Online is a service mark of Novell, Inc.

NTR2000 is a trademark of Novell, Inc.

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

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

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

Open Data-Link Interface and ODI are trademarks of Novell, Inc.

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

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

Open Socket is a registered trademark of Novell, Inc. in the United States.

Packet Burst is a trademark of Novell, Inc.

PartnerNet is a registered service mark of Novell, Inc. in the United States and other countries.

PC Navigator is a trademark of Novell, Inc.

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

Perform3 is a trademark of Novell, Inc.

Personal NetWare is a trademark of Novell, Inc.

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

Portable NetWare is a trademark of Novell, Inc.

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

Print Managing Agent is a trademark of Novell, Inc.

Printer Agent is a trademark of Novell, Inc.

QuickFinder is a trademark of Novell, Inc.

Red Box is a trademark of Novell, Inc.

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

Remote Console is a trademark of Novell, Inc.

Remote MHS is a trademark of Novell, Inc.

RX-Net is a trademark of Novell, Inc.

RX-Net/2 is a trademark of Novell, Inc.

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

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

Sequenced Packet Exchange and SPX are trademarks of Novell, Inc.

Service Response System is a trademark of Novell, Inc.

Serving FTP is a trademark of Novell, Inc.

SFT is a trademark of Novell, Inc.

SFT III is a trademark of Novell, Inc.

SoftSolutions is a registered trademark of SoftSolutions Technology Corporation, a wholly owned subsidiary of Novell, Inc.

Software Transformation, Inc. is a registered trademark of Software Transformation, Inc., a wholly owned subsidiary of Novell, Inc.

SPX/IPX is a trademark of Novell, Inc.

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

Storage Management Services and SMS are trademarks of Novell, Inc.

Technical Support Alliance and TSA are collective marks of Novell, Inc.

The Fastest Way to Find the Right Word is a registered trademark of Novell, Inc. in the United States and other countries.

The Novell Network Symbol is a trademark of Novell, Inc.

Topology Specific Module and TSM are trademarks of Novell, Inc.

Transaction Tracking System and TTS are trademarks of Novell, Inc.

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

Virtual Loadable Module and VLM are trademarks of Novell, Inc.

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

Yes, It Runs with NetWare (logo) is a trademark of Novell, Inc.

Yes, NetWare Tested and Approved (logo) is a trademark of Novell, Inc.

ZENworks is a trademark of Novell, Inc.