Edison Group, Inc

When CIFS Performance Matters

Business Strategy Report

For

Microsoft

February 10, 2006

Printed in the United States of America.

First Publication: February 2006

Produced by: Stewart Miller, Senior Analyst; Andrew Podosenin, Lead Analyst; Craig Norris, Editor; Barry Cohen, Editor-in-Cheif

Table of Contents

Executive Summary

Most networking environments today are comprised of Windows clients and servers. The majority of traffic served across these networks is Microsoft SMB, the built-in Windows file serving protocol. Windows-based servers, functioning in this type of environment, offer the best CIFS performance values. These values can be quantified with benchmarking tools such as NetBench. Real-world networking environments that are Windows-based can not be measured with performance from NFS file sharing protocols for UNIX—because such values are irrelevant in a "Windows" network.

Purpose of the Report

The goal of this paper is to measure CIFS performance in a "real-world" computing environment, as reported by the NetBench benchmark. This paper reports the file serving throughput of 64-bit Windows file servers with varying numbers of clients.

Who Should Read this Report?

This report should be read by IT Professionals, CxOs, and purchasing managers who are considering network storage solutions and who wish to understand the value of various vendor-reported performance characteristics.

General Overview of SMB and CIFS

Defining SMB and CIFS

Microsoft Server Message Block (SMB) Protocol is the "mother" environment of the Windows file serving protocol. While Microsoft continually improves SMB with each release of Windows, a subset of it called CIFS (Common Internet File System) functions as the native file transfer protocol mechanism for clients to request access to files from the network server no matter what the underlying operating system platform.

Microsoft has modified SMB on the Windows 2003 server to offer enhanced services, increased performance, and better security. CIFS is the means by which remote file systems allow groups of users to access, collaborate, and share documents or files across the LAN. CIFS is an open, cross-platform protocol based on the native file-sharing protocols that come standard in Windows as well as several other operating system environments.

Client/Server file transfer performance is very important in any Windows-based environment. The comparative metric you **need** is the "*CIFS number*." This performance value quantifies exactly how quickly you can transfer files between a Windows client and other operating system platforms.

Collaborative File Sharing

Business people will find it advantageous that CIFS enables collaboration on the LAN by establishing a remote file access protocol that is already compatible with the way in which applications share information and data on local hard drives across network file servers.

The value in CIFS is its level of:

- High performance
- Multi-user read/writer operations
- Locking
- File sharing capabilities

These features are defining characteristics of the majority of enterprise computer networks meant for mission-critical tasks.

CIFS functions over TCP/IP and uses the global Domain Naming Service (DNS) to enhance scalability for most any need. It is optimized to support slower connections and is a means to yield the best performance possible in nearly any computing environment.

Network File System Traffic

Determining the best choice of a network file system for various deployed clients involves a careful examination of the types of traffic on your network. The first step is to review network file system requirements — specifically, what features and functions they are expected to deliver.

Network file systems were developed to let computer users identify, connect, and access files stored on remotely networked computers just as if the files were stored on those users' local machines. However, it is important to note that specific requirements must first be met in order to select the best solution for your needs.

Distributed files systems involve clients, servers, and storage devices dispersed across the LAN. These network services all draw into a central data repository, for which there are multiple and independent storage devices. Performance matters, because a networked file system is implemented as part of the operating system. This means there is a dedicated operating system whose task is to manage the communication between the client and the file servers. In order to achieve the maximum level of performance, it is necessary to optimize network communications using a native file transfer protocol that is integrated into your systems to facilitate data transfer.

The most important features include:

Feature	Description
File access	File operations include open, close, read, write, and seek.
File and record locking	After a file or record is locked, non-locking applications are denied access to the file.
Safe caching, read-ahead, and write-behind	Allows read/write access to a file from multiple clients simultaneously.
File change notification	Applications can register with a server to be notified when a file or directory contents are modified.
Protocol version negotiation	When client and server first come into network contact, they negotiate the version (dialect) to be used. Different dialects can include new message types as well as changes to the field formats in other dialects.
Extended	Non-file system attributes, such as the author's name, can be

Feature	Description
attributes	added to the built-in file attributes, such as creation and modification times.
Distributed replicated virtual volumes	The protocol supports multi-volume file system sub-trees which look to clients as if they are on a single volume. If the sub-tree files and directories are physically moved or replicated, the CIFS protocol uses referrals to transparently direct a client to the appropriate server.
Server name resolution independence	Clients may resolve server names using any name resolution mechanism. Using the name resolution server DNS, for example, permits access to file systems over the LAN.
Batched requests	Multiple file requests may be grouped into a single message, in order to minimize round trip latencies, even when a later request depends on the results of an earlier one.
Unicode file names	Unicode strings may be exchanged. Unicode strings include file names, resource names, and user names.

Enhanced Communications

CIFS enables clients and servers to communicate so they can share files and printing functions. This protocol is most clearly understood through its rich set of features and functionality. The following subsections break CIFS down into its primary attributes and uses in real-world computing environments.

File Access

A file server is an important part of your business, as it provides file services to clients. The client interface involves file operations such as:

- Creation
- Deletion
- Reading
- Writing

The hardware components of a file server control the local storage devices where your files are stored. The files are retrieved in accordance with the client requests.

Concurrency control and locking are two very important features in a network file system. The file system must allow one client to update one file at any given time. The goal is to ensure that another client cannot attempt to update the same file someone else is working on. Eliminating this type of interference requires a locking mechanism to prevent conflicts and is controlled by the file system itself or through an add-on protocol. The problem with add-ons is that they add considerably to processing overhead. Windows, however, has these features "built-in" so that the processes take place that much more smoothly.

CIFS allows multiple clients to access and update the same file, but this protocol prevents conflicts by using advanced file-sharing and locking mechanisms which also permit aggressive caching and read-ahead and write-behind methods without losing cache coherency. These features also work together to support fault tolerance to resist network and server failures.

Directory Paths

When CIFS is used, there is an efficient cross-platform method of identifying SMB shared resources both internally on a Local Area Network as well as on a Wide Area Network. The first step involves the client finding the server share and then initiating the connection.

NOTE: You cannot send SMB messages until you first talk to the server. If you want to connect to an SMB server, you first must resolve either the NetBIOS or DNS name to a numerical IP address. Once you have this address, only then can you try to start and open sessions with the server.

The CIFS URL is either **smb://server/share/filename** or **cifs://server/share/filename** in order to attach to the specified file you need to transfer. While it is possible to establish such connections from a UNIX, Linux, or Macintosh platform to retrieve the desired information, Windows clients are fine-tuned to yield the best possible performance, throughput, and file transfer capabilities when establishing native CIFS connections between the Windows client and server.

Network Performance

CIFS increases performance if the Windows client can buffer file data locally. Buffering strategy increases CIFS performance: the client does not have to write information into a file server if the client knows that no other processes are accessing the data. The client can buffer read-ahead data from the file if the client knows that no other processes are writing the data. CIFS performance is quantified through NetBench test results performed by The Edison Group. Please refer to the appendix for details regarding these tests.

If you have an NTFS volume with a large number of folders or files, and a program briefly accesses each of these in turn, the I/O bandwidth used to generate the Last Access Time updates can be a significant percentage of the overall I/O bandwidth. To increase the speed of access to a folder or file, you can set **disablelastaccess** to disable updating the Last Access Time. After you use this command and restart the computer, the Last Access Time is no longer updated. If you create a new file, the Last Access Time remains the same as the File Creation Time.

Natively Supported Performance Advantages

The nicest feature of CIFS is that users can share files without having to install additional software to facilitate data transfers. For example, when running FTP you must install special client and server software to upload and download files. Internally CIFS runs over TCP/IP, but it uses the SMB protocol found natively in Windows for both file and printer access. This means that CIFS permits a host of applications and web browsers to open and share files.

If the clients of a server do not access a file by both short and long names, and there are not multiple shares on that server at that point to the same file set, you can set the "**NoAliasingOnFileSystem**" key to inform the clients that they can intelligently cache the namespace for performance improvement.

Efficiency is maintained because when CIFS is used as the protocol to make changes in any given file those changes are simultaneously saved on both the client and the server.

The mechanism used by SMB works such that a client makes a specific request, and the file server responds in kind. CIFS protocol was optimized for local subnet usage, but is widely used to access different subnets across the Internet. SMB servers make file system resources available to clients on your network. Microsoft added a host of performance and value-added features to its own SMB to fine-tune the protocol and set a standard for file transfers optimized on Windows-based networks.

File Sharing Performance

The most important performance measurement of a networked file system is the amount of time required to answer service requests. Since saving time equates to productivity in business, performance matters greatly. This most often comes down to the amount of disk-access time in combination with a small amount of processor cycles. Performance is also directly tied to the overhead attributed to the distributed file server architecture whenever a user remotely accesses a file. This value is quantified by the amount of time required to deliver the request to the server in addition to the time needed to get the response across the network back from the client. So apart from the transfer of information, information travels bi-directionally across the network *and* there is also processor overhead that runs the network communications. Achieving the best performance means optimizing the client and server to experience as little overhead as possible when transferring files. This optimization is best achieved when using CIFS in a Windows-based environment. The entire idea is to achieve maximum performance so that users experience as little wait time as possible. This goal of what performance means to you essentially equates to increased business productivity. Productivity falls when users are waiting for the client and server to effectively communicate across the network in order to retrieve a file.

CIFS protocol was designed to operate well over slow network connections. It provides improved performance for users who access the network with slow modem connections. Regarding performance, the original SMB protocol was not meant to work over slow network connections (i.e., dialup). Several contributing factors make the standard SMB protocol a less efficient protocol than CIFS. Original SMB currently offers very little support for client-side caching[1]. Performance decreases because this version of SMB has a tendency to open and close a file for multiple writes. Furthermore, original SMB has no ability to connect to a different server to take advantage of better performance from a server that is geographically closer and/or offers greater speed.

Microsoft (MS) SMB carries a richer file sharing feature set than CIFS. MS SMB enables the sharing of both directories and files. CIFS runs in Windows to allow access to files and sharing of printer access. This means CIFS will allow applications (not just web browsers) to open and share files across the LAN.

Speed is an important metric, because you can essentially get more work done when information is readily available when you need it.

Why CIFS performance excels is best explained by measuring file sharing performance for Windows clients. NetBench is the *current* standard performance test that measures file transfer performance metrics using the CIFS protocol. Microsoft SMB has better CIFS performance results because of its enhancements in both file sharing performance and security. The Edison's Group test of file sharing performance using NetBench shows the incredible performance that Windows 2003 Server is capable of producing. Microsoft's enhancements in SMB are illustrated best through the speed of file transfers.

CIFS effectively becomes the standard for current file sharing operations in a Windows-based organization. File sharing operations can network most effectively, as organizations can move their existing file sharing structure onto the network, producing a most effective extranet.

[1] Microsoft SMB continues to be enhanced with additional features expected in Windows Vista™.

Security

Security is critical when doing business today. CIFS employs a security model designed to protect business information from prying eyes. Each server provides a set of defined resources available to clients on the network. Shared resources can be located in a either a directory tree, named pipe, or printer. Clients do not see the server as having any storage or service dependencies on any other servers; the server appears to be the main resource for any file it needs to access.

CIFS servers permit anonymous transfers as well as secure/authenticated access to named files. Administrative policies for both file and directory security is also easier to support. SMB was originally designed to work in LANs, but lacks a strong set of security features. CIFS, by contrast, was built more securely with both encryption and secure authentication functionality. CIFS also has a more flexible naming schema, allowing users to name a CIFS file server using the name of the computer, DNS, or IP address.

SMB requires server authentication of users before any file can be accessed. Since each server authenticates its own users, the client must send authentication information to the server prior to allowing a user to access any information.

The two methods by which SMB enforces server security involve:

- Share-Level Security
- User-Level Security

A share-level server makes some directories on the disk available to the user, who may require a password to gain access. This means that any user on the network who knows the name of the server, resource, and a valid password can access the information.

Share-level security involves servers that use different passwords for the same shared resource. This means that there are different passwords that allow for different levels of access to various types of information.

A user-level server makes some directories on the disk available, but also makes certain that the client provides both a username and password in order to gain access. The user-level servers are often best even over shared-level services, due to the fact that companies find user-level servers far simpler to manage on a per-person basis.

An SMB/CIFS server keeps an encrypted form of the client's password. So, for a user to gain authenticated access to server resources, the server must first send a challenge to the client, to which the client responds in a way that proves it knows the appropriate password.[2] CIFS is primarily used for Windows shares and providing other clients with shared printer access. SMB is more commonly used to mount file shares that look like locally connected network drives.

Scalability

Windows is tuned for maximum system performance when it comes to CIFS, because of the protocol's tight integration with the operating system. When most of the traffic on networks travels across Windows-based machines, you can expect to achieve the ultimate levels of performance.

The SMB_COM_FLUSH command makes sure that all data and allocation information for the corresponding file has been written to stable storage. A response is not sent to the client until the writes are complete. However, two factors may contribute to unnecessarily poor server performance:

- Many client programs flush files more frequently than is required. Client programs should flush files only when this is necessary for data consistency.
- Too many data flushes slow the performance of the file server because the flushes interfere with the server's ability to optimally schedule disk activity. Programs that are running on the server do not know that it is sufficient to leave the data in the memory of the file server; however, if the file server is sufficiently reliable, it may be acceptable to keep file data stored there. In this case, the memory on the file server is considered *stable storage*, but the programs are not aware of it.

Filenames

CIFS uses filenames in any character set. This means you are not limited to character sets designed for English or Western European languages. With respect to Global File Names, users need not mount remote file systems, but can refer to them with names that can be virtually anywhere on the LAN. The result is that the Uniform Naming Convention (UNC) can support filenames so that a drive letter does not need to be indexed before accessing files.

CIFS does not make HTTP obsolete, but complements it. CIFS yields significantly better performance for both file sharing and file transfer than protocols such as FTP. Alternatively, HTTP is better for some file sharing operations so that external users can be guided to specific files.

[2] For example, when transferring files through the Microsoft SMB protocol to a Unix file system client; Windows 2003 Server digitally signs each packet before a share can be mounted or executing a file transfer—a feature that is not standard in the generic SMB or CIFS protocol.

NFS

NFS (Network File System) was developed by Sun Microsystems as the file system for its networked computers. NFS is a common protocol used in UNIX platforms for sending files across a network. A web version of the NFS distributed file system from Sun is called, "WebNFS" which permits users to access information more quickly than HTTP, but also downloads multiple files in one connection.

The main point in either CIFS or NFS is that it provides you with an essential level of fault tolerance for large downloads that lose their connection in the middle of a file transfer. You will see a number of diehard UNIX techies telling you that NFS provides you with greater performance and capabilities.

But, let us examine that claim of performance in greater depth. NFS does transfer files better than HTTP; however, is performance not tied to the efficiency of your network? The optimization of file transfer rests expressly on the server's ability to natively serve information in the most efficient method possible. If your networks are primarily based on UNIX servers, then NFS may indeed provide you with greater performance.

Windows is the most commonly used Operating System that accounts for the majority of end-user computing traffic on corporate networks. CIFS is the best choice for operating environments where the majority of traffic is generated by Windows. CIFS has several advantages over NFS when your "real life" network consists mainly of Windows-based OS environments. CIFS offers a better level of native optimization at work between your Windows server and client that effectively speeds up the transfer of files. Windows was designed from the ground up with the goal of providing the most efficient transfer medium for information possible, and supports a traffic pattern that favors CIFS over NFS.

When you compare CIFS and NFS, it is important to consider what comprises the majority of your network traffic. CIFS is more important when dealing with networks where traffic is primarily Windows-based. NFS is only an important metric when the majority of your traffic is UNIX or Linux-based. Essentially, CIFS is a better file sharing performance gauge than NFS in networks composed mainly of Windows machines.

A true comparison between CIFS and NFS involves two main factors:

- Examining what *type* of traffic is being generated in your file sharing environment (Windows or UNIX).
- Quantifying performance using NetBench in such a manner that networks with mainly Windows traffic have performance measured by comparing CIFS metrics.

NetBench Performance Metrics

NetBench is *the* most commonly used standard benchmark used to test CIFS performance. It provides you with a tangible comparison between different computing devices. By asking for a vendor's CIFS performance NetBench measurement, you can effectively determine your performance metric.

Edison Group was hired by Microsoft to measure the file server performance of Windows 2003 64-bit Server. The stress test was performed with the industry-standard NetBench 7.03 software with multiple client nodes running in a non-blocking network infrastructure. The test was performed at Edison Group Labs in New York, NY. The server, the client, and the networking hardware were rented from a third party. The server and the client operating systems were purchased from a mail-order store. Microsoft employees were not permitted to access the Edison Group Labs during the test. The server has passed the Windows 2003 Enterprise Server compatibility test and was tuned in accordance with the Performance Tuning Guidelines from Microsoft[3]. The standard test suite was extended to allow 200 clients to run. The observed peak throughput was 4,123 Mbit/s for 64-bit Windows Server 2003 SP1, and 3,994 Mbit/s for 64-bit Windows Server 2003 R2. The maximum performance point was reached as indicated by the shape of the performance curve. The stress-test was performed three times to ensure the stability and reproducibility of results. In order to get the more accurate number, the stress test was repeated around the global maximum of the curve with a smaller step.

Performance Analysis Methodology

When working in an environment where traffic is primarily composed of Windows-based traffic, NetBench is commonly accepted as the standard. Windows CIFS file sharing solutions use NetBench to quantify their peak performance. Even when looking at a network attached storage appliance, always ask for your CIFS performance measured using NetBench — only then will you see how important CIFS performance really is.

Performance was measured by the standard NetBench test suite. NetBench benchmark suite measures file server performance as the number of bytes transferred to and from the server per second. NetBench suite tests a file server by generating the controlled number of requests from the managed load-generating nodes. Each node can emulate

[3] Performance Tuning Guidelines for Windows Server 2003. Microsoft Corporation. October 2003. http://www.microsoft.com/windowsserver2003/evaluation/performance/tuning.mspx

several file-accessing clients. NetBench test site was configured to incrementally generate the load on the system until the maximum performance is reached.

Test Lab Design

Stress-Testing Software

NetBench version 7.03[4] was used for load generation. This is the most recent stable version of NetBench. Standard NetBench tests were used with modification of the mixes to allow for over 60 clients. Additional client directories for these clients were created on the server.

Stress-testing Nodes and Client Tuning Parameters

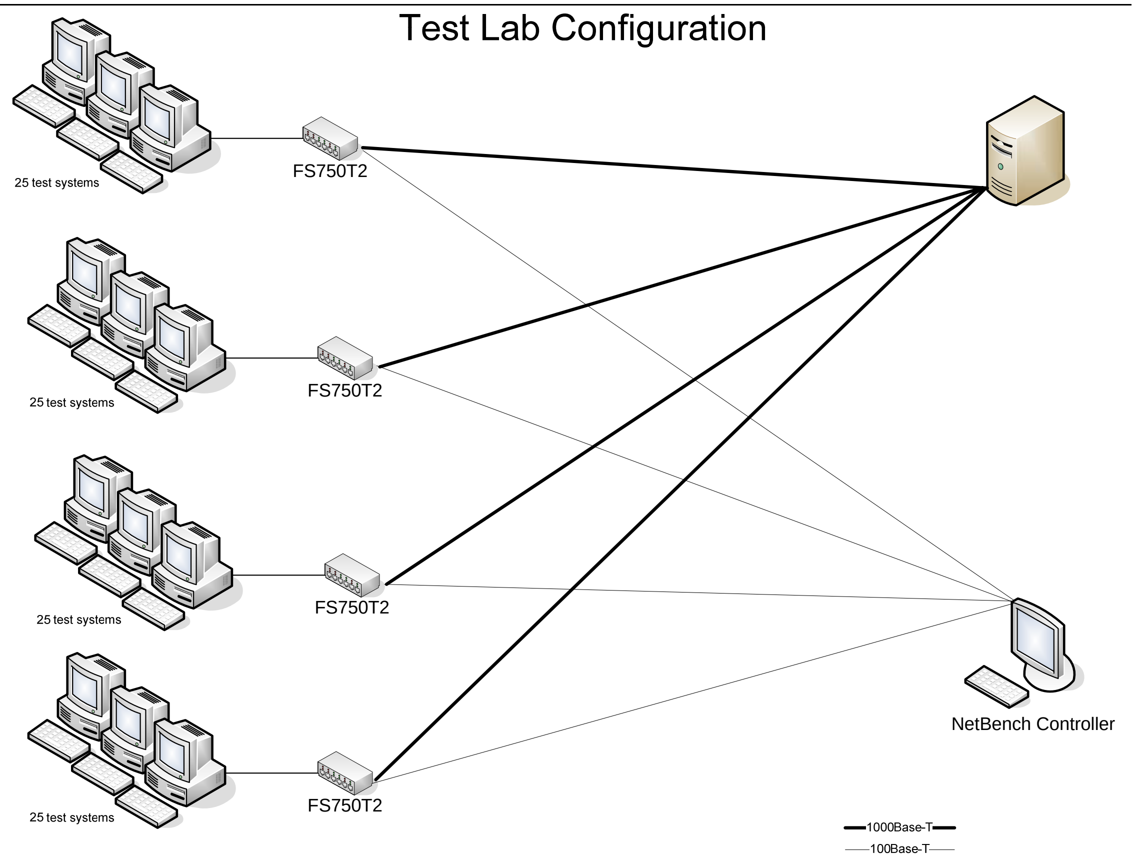
A total of 100 stress-testing nodes were used to generate the load. The nodes had the following hardware configuration:

- Generic workstation with Biostar M7VIG 400 mainboard
- Sempron 2200+ (1.5 GHz)
- 512 M of AENEON PC3200 DDR1 RAM
- 40 GB 7,200 rpm ATA 133 Maxtor hard drive
- On-board VIA VT6103 10/100 Mbit NIC
- One NetBench controller station was used with the hardware and OS identical to those of a stress-testing node.

Networking Topology

The network was composed of four NETGEAR FS750T2 48-port 10/100 Mbit switches with two Gigabit uplinks. The server and the NetBench controller were connected to each of the four switches. The most recent firmware was installed on all the switches. The complete network diagram of the test lab is presented below.

[4] http://www.veritest.com/benchmarks/netbench/default.asp


Test Lab Configuration
25 test systems
FS750T2
25 test systems
FS750T2
25 test systems
FS750T2
25 test systems
FS750T2
NetBench Controller
1000Base-T
100Base-T

Preliminary Tests and Assumptions

Prior to running each test suite, the NetBench software was reinstalled and the data volume was reformatted to avoid any residual data from previous test runs. Upon completion of each successive test, the file server, the controller, and the stress-generating nodes were rebooted.

Accuracy and Variability of Results

All the tests were performed with fine-tuning of the server and the client based upon Microsoft guidelines for tuning file transfer performance. We believe we were able to get the maximum performance out both Windows Server 2003 SP1 and Windows Server 2003 R2. Though R2 exhibited slightly lower performance numbers, the difference between the two versions was very slight.

Each measurement was taken three times, with approximately 40 points difference between the minimum and the maximum results at the peak performance points. Therefore, the obtained results are 99.5% accurate and can be reported as 4,120±20 and 3993±20. It is probable that additional tuning will be possible as Windows Server 2003 R2 matures as a platform.

The points appearing to be the global maxima in each case are higher than the neighboring points by more than 20 units. Additionally, the visual (and numerical) interpolation of the obtained graphs demonstrates that we have a curve with one global maximum located at the coordinates stated in our report.

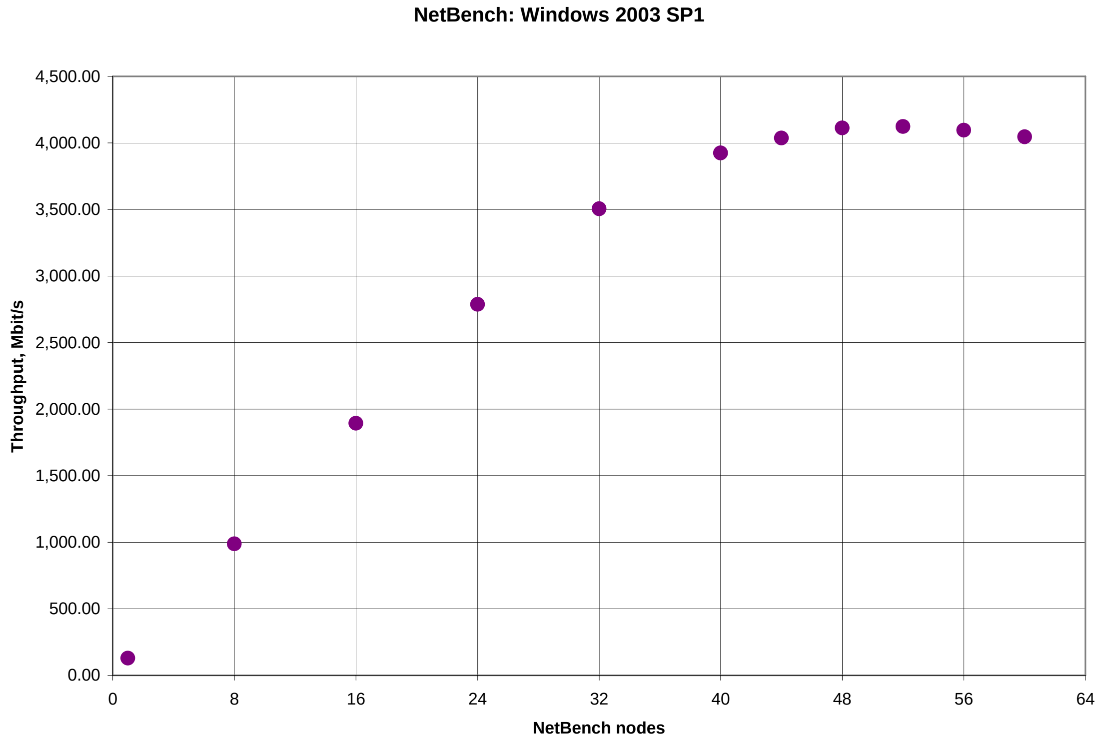
NetBench Test Values

The following results were obtained for 64-bit Windows 2003 Server SP1

Nodes	Throughput, Mbit/s
1	129.54
8	987.12
16	1,893.99
24	2,786.90
32	3,505.36
40	3,924.79

44	4,037.36
48	4,112.86
52	4,123.73
56	4,095.39
60	4,045.60

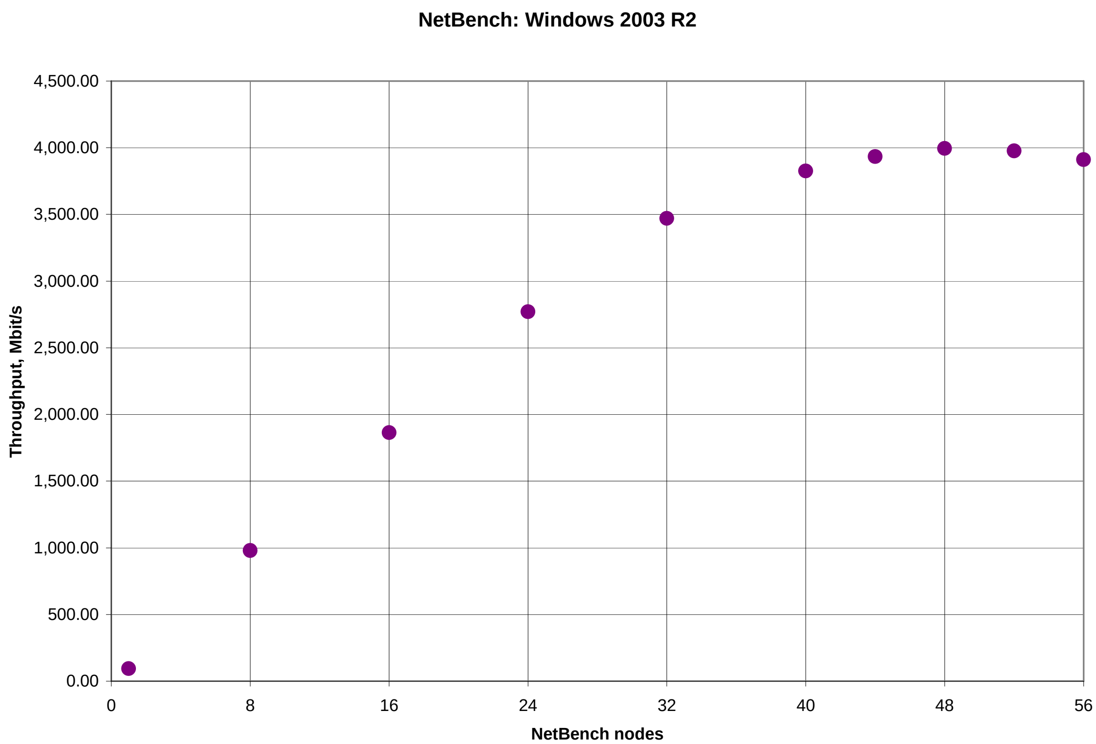
The results are displayed on the graph below.



The following results were obtained for 64-bit Windows 2003 Server R2

Nodes	Throughput, Mbit/s
1	94.02
8	980.20
16	1,864.05
24	2,770.73
32	3,470.38
40	3,825.58
44	3,932.94
48	3,993.63
52	3,975.94
56	3,911.00

The results are displayed on the graph below.


NetBench: Windows 2003 R2
Throughput, Mbit/s
4,500.00
4,000.00
3,500.00
3,000.00
2,500.00
2,000.00
1,500.00
1,000.00
500.00
0.00
0
8
16
24
32
40
48
56
NetBench nodes

Conclusion

Most traffic *is* CIFS in a Windows-based networking environment.

The simple fact is that most traffic traveling across the LAN occurs between Windows clients and servers. There is simply *no contest* between the importance of CIFS performance and NFS performance. Computer networks run Windows. Windows runs CIFS natively — and there are significant performance benefits because CIFS is an integral part of the Windows environment. This means that NFS performance figures are moot when traffic is mostly generated by Windows clients

The following three factors sum it up best:

- CIFS scales most effectively on standard Windows platforms.
- CIFS grows proportionally to meet your needs today.
- CIFS scales extremely well to meet your needs in the future as your file sharing and storage needs grow.

Performance matters because it is the governing factor in fostering productivity for your business operations. Spending a great deal of money on ultra-fast storage devices means nothing unless you have a file system like CIFS to take advantage of that speed. When used in combination with the integrated performance features of using a Microsoft Server, CIFS is an important means of leveraging your investments in storage as well as in your file serving infrastructure.

In summary: if you are running in a Windows world, file transfers demonstrate the greatest efficiency and the absolute best performance with CIFS.

Appendix

File Server Hardware and Tuning Parameters

Windows 2003 64-bit Enterprise server was installed using default settings as a workgroup server with a file server role. The on-board storage was configured as RAID 1 with 30 GB OS volume (drive C) and a 42 GB application volume (drive D). The external disk enclosure was configured as four RAID 0 volumes, two per channel on the dual-channel RAID controller with 12 72G 10 K RPM drives. All the drives were formatted with an NTFS partition. Each set of 25 clients were configured to access a separate logical volume of the array.

The following changes were made to the server:

- Server set to maximize throughput for file sharing and to boost the background applications. NICs were set to the following:
  - NIC Speed – 1 Gbit
  - Spooler service was disabled
- The RAID system was formatted with 128 KB allocation unit size.
  Raid Controller settings:
  - Rebuild priority – Low
  - Expand priority – Low
  - Accelerator ratio – 50% read, 50% write

Additional Information

Edison Group has published the complete results of its tests in the document library at http://www.theedison.com (A direct link to the library is: http://www.theedison.com/index.php/articles/c51/) Registration is required.

Additional information is available from the Edison Group:

Edison Group, Inc.

89 Fifth Avenue
Seventh Floor
New York, NY 10003
info@TheEdison.com