SPEC SFS Performance on eServer pSeries™ Systems

Agustin Mena III

February 5, 2002

Abstract

A common use of UNIX® servers is in fileserving environments, specifically those using the Network File System (NFS) protocol [1, 2]. SPEC SFS 3.0 is an industry standard benchmark for measuring and comparing NFS server performance across a variety of file server vendors' systems. Based on recently published SPEC SFS 3.0 results, the IBM® eServer pSeries™ 690 running AIX® 5L Version 5.1 proves to be an excellent choice for customers interested in a general-purpose UNIX server with outstanding performance in a large-scale NFS environment. The p690 results published in December 2001 are the highest of any SMP system in the industry. Also, the recently published (January 2002) IBM eServer pSeries 660 6M1 8-way results showcase this system as an excellent midrange NFS server. This white paper discusses the benchmark, the system configurations used, and details of the results and how they stack up against the competition.

The SPEC SFS 3.0 Benchmark

SPEC (Standard Performance Evaluation Corporation) SFS (System File Server) 3.0 is the most recent version of the industry standard SPEC benchmark which provides a means of comparing NFS server throughput and response time performance across different vendor platforms [3]. The benchmark is client-independent and vendor neutral. The results must conform to a set of run and disclosure rules. It is this set of rules that ensure that customers have a reasonable standard to compare and contrast the results across different vendor systems. Prior to being officially posted on the public SPEC website (http://www.spec.org), the results are validated via a peer review process by representatives of SPEC SFS committee member companies. SPEC does not allow SFS 3.0 estimates or unofficial results to be publicized.

SPEC SFS 3.0 replaced the SPEC SFS 2.0 benchmark in September 2001. SFS 2.0 was withdrawn by SPEC in June 2001 due to several bugs discovered in the benchmark which compromised accurate comparisons of many of the results [4]. Specifically, the working set was found to be smaller than intended by the benchmark designers, and the file access distribution was found to be sensitive to the number of load-generating processes. In addition to fixes for these bugs, SFS 3.0 includes the following improvements over SFS 2.0:

- More robust validation of NFS operation handling by the NFS server.
- Reduced requirements for client memory size.
- Support for Linux and FreeBSD clients.
- New tools for results submission.
- Improved documentation.

SPEC SFS 2.0 replaced the SFS 1.1 benchmark often called LADDIS. The evolution of server technology and changes in customer workload characteristics necessitated updating SFS 1.1, which was released in November 1994. These are some of the more significant enhancements introduced in SFS 2.0, which was released in December 1997:

- Support for both NFS version 2 and NFS version 3 (see [5] for information on the version differences). SFS 1.1 only supported NFS V2.
- Support for both UDP and TCP network transports. SFS 1.1 only supported UDP.
- Modified the workload NFS operation mix to reflect more up-to-date customer environments. For instance, the SFS 1.1 benchmark was considered to be too write-heavy, so

the percentage of writes was dropped from 15% to 7% in the SFS 2.0 V2 workload. See Table 1 for the full list of operation percentages by workload.
- Doubled the test file set size (to 10 MB per ops/sec requested load) as compared to SFS 1.1, but kept the absolute amount of data accessed the same.
- Changed the test file set to include a range of file sizes (see Table 2), as opposed to using a single file size (136 KB) as in SFS 1.1. As with the NFS op mix modification, these changes were based on data access patterns seen in customer environments.

It is important to emphasize that SPEC SFS 3.0 consists of two separate workloads, one for NFS V2 and one for NFS V3, which report two distinct metrics, SPECsfs97_R1.v2 and SPECsfs97_R1.v3, respectively. The results from these workloads gathered on the same machine cannot be compared against each other. In general, SPECsfs97_R1.v3 throughput will be less than SPECsfs97_R1.v2 throughput on any given system. This does *not* mean that the NFS V3 performance is worse than NFS V2 performance on that platform.

The SPECsfs97_R1 metrics consist of a throughput component and an overall response time measure. The throughput (measured in operations per second) is the primary component used when comparing SFS performance between systems. The overall response time (average response time per operation) is a measure of how quickly the server responds to NFS operation requests over the range of tested throughput loads.

Details on execution characteristics of the benchmark, and on the system configurations used in generating the p690 16-way and the p660-6M1 benchmark results, are provided in the next two sections.

Benchmark Execution Characteristics

The SFS 3.0 benchmark generates an increasing load of NFS operations against the server and measures the actual throughput (in ops/sec) and average response time (in msec/op) achieved. Official test runs are required to consist of at least 10 evenly spaced load points between 0 and some chosen value, usually the maximum expected throughput.

The benchmark code runs on load generators (“clients”) and generates the necessary RPCs (Remote Procedure Calls) to simulate an NFS client. Therefore, having NFS client code installed on the clients is not required. Each client generates an equal portion of the overall requested load. The filesystems used by each client are specified in a benchmark configuration file. The benchmark run rules require that the I/O load be evenly spread across all the disk storage adapters in the system. The run rules also require that the network load be evenly spread across all the network segments over which the server and clients communicate.

The benchmark consists of three phases:

1. Initialization phase. During this phase, the test filesystems are populated with directories/files based on the amount of load requested for the particular point in the run.
2. Warmup phase. Here the actual workload is executed for 300 seconds in an attempt to have the system reach a steady state so that benchmark start-up costs and variability do not affect the throughput and response time measurements.
3. Run phase. It is during this phase (also 300 seconds in duration) that the actual throughput and response times are measured.

System Configuration

The system configuration used in running the SFS 3.0 benchmark consists of the NFS server and several clients communicating with the server over one or more network segments. The specific setup used to achieve the p690 16-way results is depicted in Figure 1.

It is important to note that the server system is configured such that CPU saturation becomes the performance-limiting factor for the benchmark. Every attempt is made to remove all other system bottlenecks by using sufficient memory, network adapters, disk adapters, and disks. Therefore, an SFS 3.0 server configuration may not be typical of what customers would use in a real-world environment. For example, while customers typically decide on the number of disk drives to buy based on their space needs, the benchmark needs are based primarily on the number of drives needed to sustain the overall IOPS (I/Os per second) requirement given the NFS ops/sec throughput that we want to achieve. Disk capacity must be considered in setting up to run SFS 3.0 since the benchmark creates 10 MB of data per NFS op/sec, but the overriding requirement is sufficient disk arms to maintain the needed IOPS. For instance, in the p690 SPECsfs97_R1.v2 configuration, we used a total of 512 SSA disks to house the 448 test filesystems. 64 volume groups (VGs) were defined, with 8 disks per VG: 1 disk contained the filesystem logs for that VG, and each of the other 7 contained 1 filesystem.

We used what may seem like an extraordinary number of disks. In fact, iostat data gathered during internal (unofficial) p690 16-way benchmark runs using NFS V2 shows that all the disks were only about 30% busy at the highest NFS ops/sec throughput that we achieved. However, the large number of disks is needed in order to ensure good response times, and to comply with the benchmark run rules (e.g., uniform load distribution across all disk adapters). Using the SSA fast-write cache option, as we did during the published result runs, helps to reduce the number of disks required. During other internal runs with fast-write disabled, iostat data showed that compared to runs with fast-write enabled, disks housing the filesystem data were 25% busier, and disks housing the filesystem logs were twice as busy.

As with the number of disks and disk adapters, the number of network adapters used for the p690 16-way measurement is more than the minimum required to support the peak NFS V2 throughput. At peak, only about 14000 SPECsfs97_R1.v2 ops/sec worth of network packets were flowing through each Gigabit Ethernet adapter. Subsequent internal benchmark runs have shown that each adapter can handle around 18000 ops/sec of traffic with no negative effect on NFS throughput or response time.

The p660-6M1 achieved a peak SPECsfs97_R1.v2 throughput which was roughly half that of the p690. Thus, we used a hardware configuration half of that of the p690 in terms of network adapters, disk adapters, disks, and clients. For the p660-6M1 we used: 16 SSA adapters, 256 SSA disks, 4 Gigabit Ethernet adapters, and 4 44P-270 4-way clients.

Besides tuning the hardware configuration, choosing an appropriate software configuration and the proper values for AIX tunable parameters is also key to achieving good SFS 3.0 performance. The benchmark runs on the p690 and p660-6M1 were done using the AIX 5L V5.1 64-bit kernel, and the test filesystems were configured as Enhanced Journaled File System (also known as JFS2) filesystems. Here are the AIX parameters which were tuned on the p690 16-way server during the benchmark runs:

- Gigabit Ethernet adapter **jumbo_frames = yes**

  Setting this attribute on the Gigabit Ethernet adapter allows the use of jumbo frames (9000-byte MTU), in place of the default 1500-byte MTU. In environments where the majority of NFS packets are large, using the larger MTU size can result in more efficient server CPU utilization due to fewer trips through the UDP packet fragmentation and reassembly code. In the SFS 3.0 workloads, the typical NFS packet size is relatively small, but using the larger MTU size still has a slight positive effect on benchmark performance.

- **vmtune -P 100**

  This vmtune parameter controls the maximum amount of memory to be used for file caching. The value is expressed as a percentage of the total real-memory page frames in the system. The value of 100 effectively allows all system memory to be used for caching file pages. In customer environments where the server is used for multiple applications, not just NFS fileserving, the default of 80% may be more appropriate. In the SFS environment that was used for the p690 tests, setting this tunable was probably not required since: (1) the SFS test filesystems were configured as JFS2 filesystems, (2) the amount of memory (128 GB) in the system was more than twice the amount required to cache the benchmark working set for the SPECsfs97_R1.v2 peak throughput that was achieved. JFS2 uses client segment (not persistent segment) memory for caching file pages. The maximum amount of client segment memory to be used for file caching may be controlled via the **vmtune -t** parameter. See the discussion below on how this parameter was tuned for the p660-6M1 runs.

- **nfs_server_clread = 0**

  Setting this nfso nfs_server_clread parameter to 0 disables the use of an aggressive read-ahead strategy by the NFS server in conjunction with the local (exported) filesystem. Due to the random access nature of the SFS workload, disabling read-ahead results in less unnecessary I/O activity (by avoiding page-ins of data not used by the benchmark). But, it can result in less than optimal performance in environments where sequential reads of large NFS files are prevalent.

- **nfs_server_base_priority = 80**

  Setting this nfso parameter fixes the priority of the nfsd threads. By default, the nfsd daemons run with floating priority, and thus their priorities change as they accumulate CPU time. In an environment where the machine is dedicated to NFS fileserving, fixing the priority of the nfsd threads may be advisable. We've seen that, in general, this results in improved NFS operation response times.

- **nfs_max_threads = 1024**

  This nfso parameter specifies the maximum number of NFS server threads that are created to service incoming NFS requests. The number of active nfsd threads is adjusted periodically via a load balancing algorithm which is based on the rate of NFS requests. Too low a value may yield poor performance if there are not enough threads to keep the CPUs busy working on NFS operation requests simultaneously. On AIX 5L V5.1 (with APAR IY22854) and beyond, specifying too high a value (i.e., much higher than the number of threads required to adequately satisfy the multiple simultaneous requests from the clients at peak load) results in no performance penalty. However, on previous AIX 5L (and all AIX 4.3) levels, the load balancing code is not optimized, and specifying too high a value of nfs_max_threads may result in wasted CPU utilization due to unnecessary dispatching of threads.

- **nfs_socketsize = 800000**

  This nfso parameter sets the send and receive queue sizes of the NFS server UDP socket. The socket is used for receiving NFS client requests, and the sizes can be adjusted so that the server is less likely to drop packets under heavy load. The value must be less than that chosen for the **sb_max** parameter (settable via the **no** command).

All of the above AIX parameters were also tuned on the p660-6M1 8-way server during its benchmark runs. In addition, one other parameter value was adjusted:

- **vmtune -t 100**

  This parameter controls the maximum amount of client segment memory that can be used for file caching (e.g., by Enhanced JFS). It is defined as a percentage of the lower of the following two values: (1) 83% of system memory (this is a limit imposed by the Virtual Memory Manager), or (2) the maxperm value (set via **vmtune -p**). In the p660-6M1 test environment, with maxperm set to 100%, the amount of client segment memory available for file caching was 83% (i.e., around 23.2 GB) of the total available system memory (28 GB). The amount of memory in the p660-6M1 was barely enough to cache the benchmark working set for the SPECsfs97_R1.v2 peak throughput that was achieved. More memory would have been desirable to provide us some headroom to ensure that we were not incurring more page-ins than required due to misses in the file cache.

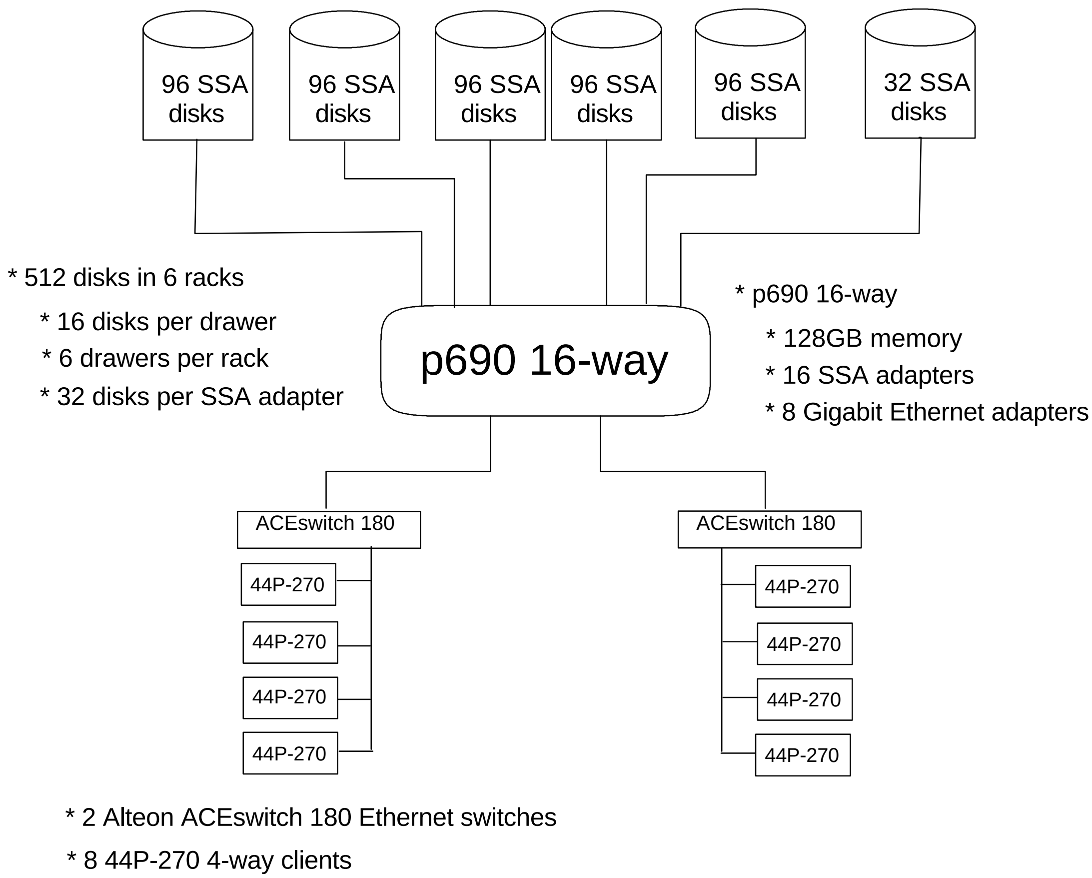

Figure 1. p690 16-way SFS configuration.

Key System Advantages

NFS fileserving workloads are, in general, data-movement intensive and not necessarily compute-intensive. Therefore, performance of these workloads is very dependent on how efficiently data is transferred between the processors, network subsystem, system memory, and the disk subsystem. High processor clock speed (to enhance raw compute power) must be accompanied by good performance traits in all of the other subsystems. The p690 and p660-6M1 exhibit excellent characteristics in all of these areas, thus enabling them to provide outstanding NFS fileserving performance.

The eServer pSeries 690 is IBM’s most powerful UNIX server to date, appropriate for a variety of high-end customer environments and applications. The p690 system architecture provides exceptional flexibility in how the processor, memory, and I/O subsystems are configured. As mentioned in the previous section, however, workload characteristics must be well understood in order to configure a p690 for optimum performance [6].

While less complex, the eServer pSeries 660-6M1 midrange server also allows a myriad of configuration options that must be considered when sizing it for a particular customer application. For instance, in order to make the best use of the available bandwidth, both of the memory riser cards should be used, and the memory should be evenly distributed across both cards [7].

In addition to appropriately configured hardware and properly tuned software parameters, good NFS server performance is also dependent on software that by default is optimized to take advantage of the underlying strengths of the hardware. Therefore, it is important to note that several important enhancements in AIX 5L Version 5.1 were instrumental in achieving the impressive p690 and p660-6M1 SFS 3.0 results. Some of the key improvements were within the JFS2 code. JFS2 is a new journaling filesystem introduced in AIX 5L which takes advantage of the larger kernel address space offered by the 64-bit kernel [8]. In general, it is characterized by design choices (e.g., B-tree directory organization) that make it much more scalable than JFS. These design choices allowed for some relatively painless scaling of data structure sizes within JFS2 in AIX 5L V5.1 which resulted in better caching of file metadata and subsequent improvements in SFS performance. In addition, there were code changes within NFS to improve its scalability on SMP systems by reducing contention on certain key locks, and by optimizing the code which handles incoming NFS requests and dynamically adjusts the number of nfsd threads in the system based on load.

Interpreting the Results

Tables 1 and 2 contain a list of the current SFS 3.0 results for the NFS V2 and V3 workloads. For the most up-to-date list, go to the SPEC SFS 3.0 web page (http://www.spec.org/osg/sfs97r1).

Understanding the system architectures is important when comparing different servers. For instance, while IBM currently occupies the top spots in SPEC SFS 3.0 performance, the highest results published on SPEC SFS 2.1 (see http://www.spec.org/osg/sfs97) were achieved by a 16-node (non-failover) cluster system. Each node is a relatively independent, dedicated fileserver with the sole purpose of moving data between the network and the storage subsystem to service NFS requests. Clusters are loosely-coupled systems with little communication between nodes, thus allowing them to be extremely scalable. Systems such as those are efficient fileservers. However, unlike the IBM p690 and p660-6M1 SMP systems, they are not general-purpose UNIX servers appropriate for hosting a variety of customer applications.

Some vendors choose to publish SPEC SFS results for a specific system configuration across all combinations of NFS protocol and network transport (i.e., V2 UDP, V2 TCP, V3 UDP, V3 TCP). These results show that, in general, peak throughput achieved on a system using the same NFS protocol can be up to 5% lower when running over TCP instead of UDP (due to communication protocol overhead). They also show that for a given system using the same network transport, peak throughput achieved on the SPECsfs97_R1.v3 workload is typically 50% to 75% of that achieved on SPECsfs97_R1.v2. These are distinct workloads with different NFS operation mixes. Therefore, it is important to emphasize once again that this does not mean that NFS V3 performance is worse than NFS V2 performance on that particular platform.

While SFS 3.0 provides one impartial method of comparing NFS server performance between different machines, you must also keep in mind the characteristics of the workloads. Note that from the server perspective, the SFS read/write accesses are mostly random, and network packet sizes are relatively small. Therefore, the benchmark is not representative of environments where, for instance, the vast majority of NFS traffic consists of very large files being read/written

Table 1. SPECsfs97_R1.v2 Throughput Results

Company	System	Network Protocol	#CPUs	Throughput
IBM Corporation	**IBM eServer pSeries 690 Turbo**	**UDP**	**16**	**111,687**
IBM Corporation	**IBM eServer pSeries 660 Model 6M1**	**UDP**	**8**	**53,745**
Hewlett-Packard	HP server rx4610	UDP	4	22,284
Auspex Systems, Inc.	NS3010 (RAID0, NetOS 4.2)	UDP	2	20,113
Hewlett-Packard	HP server rp7400	UDP	4	19,934
Auspex Systems, Inc.	NS3010 (RAID5, NetOS 4.2)	UDP	2	19,034
Hewlett-Packard	HP server rx4610	TCP	4	18,369
Hewlett-Packard	HP server rp7400	TCP	4	17,274
Auspex Systems, Inc.	NS3010 (RAID5, NetOS 4.2)	TCP	2	16,063
Procom Technology Inc.	NetFORCE 3500	UDP	1	14,232
Hewlett Packard	HP surestore NAS VA 105J	UDP	2	14,044
Hewlett-Packard	HP surestore NAS VA 105L	UDP	2	13,587
Hewlett-Packard	HP surestore NAS VA 105A	UDP	2	11,334

Table 2. SPECsfs97_R1.v3 Throughput Results

Company	System	Network Protocol	#CPUs	Throughput
IBM Corporation	**IBM eServer pSeries 690 Turbo**	**UDP**	**16**	**61,120**
Network Appliance, Inc.	F880c Failover Cluster (UDP, 100x36GB FCAL, 2xGbE)	UDP	4	30,135
IBM Corporation	**IBM eServer pSeries 660 Model 6M1**	**UDP**	**8**	**29,962**
Network Appliance, Inc.	F880c Failover Cluster (TCP, 100x36GB FCAL, 2xGbE)	TCP	4	28,596
Network Appliance, Inc.	F840c Failover Cluster (UDP, 84x36GB FCAL, 2xGbE)	UDP	2	20,520
Network Appliance, Inc.	F840c Failover Cluster (TCP, 84x36GB FCAL, 2xGbE)	TCP	2	19,007
Network Appliance, Inc.	F880 (UDP, 50x36GB FCAL, GbE)	UDP	2	17,531
Hewlett-Packard	HP server rx4610	UDP	4	16,791
Network Appliance, Inc.	F880 (TCP, 50x36GB FCAL, GbE)	TCP	2	16,685
Hewlett-Packard	HP server rp7400	UDP	4	15,202
Hewlett-Packard	hp server rx4610	TCP	4	14,848
Network Appliance, Inc.	F820c Failover Cluster (UDP, 52x36GB FCAL, 2xGbE)	UDP	2	14,683
Hewlett-Packard	HP server rp7400	TCP	4	14,104
Network Appliance, Inc.	F820c Failover Cluster (TCP, 52x36GB FCAL, 2xGbE)	TCP	2	13,777
Network Appliance, Inc.	F840 (UDP, 42x36GB FCAL, GbE)	UDP	1	11,873
Network Appliance, Inc.	F840 (TCP, 42x36GB FCAL, GbE)	TCP	1	10,909
Network Appliance, Inc.	F810c Failover Cluster (UDP, 42x36GB FCAL, 2xGbE)	UDP	2	9,789
Network Appliance, Inc.	F810c Failover Cluster (TCP, 42x36GB FCAL, 2xGbE)	TCP	2	9,786
Network Appliance, Inc.	F820 (UDP, 26x36GB FCAL, GbE)	UDP	1	8,350
Network Appliance, Inc.	F820 (TCP, 26x36GB FCAL, GbE)	TCP	1	8,020
Network Appliance, Inc.	F810 (TCP, 18x36GB FCAL, GbE)	TCP	1	4,974
Network Appliance, Inc.	F810 (UDP, 18x36GB FCAL, GbE)	UDP	1	4,967
Traakan, Inc.	RainFiler(1-Node,4x36GB,TCP)	TCP	1	2,211
Traakan, Inc.	RainFiler (Failover,1-Node,4x36,TCP)	TCP	1	1,957

sequentially. As mentioned earlier in the System Configuration section, choosing the appropriate benchmark system configuration (both hardware and software) is key to achieving good SFS 3.0 performance. The discussion in that section illustrates the importance of understanding a customer's specific fileserving requirements. The SFS 3.0 results provide only one means of comparing NFS server performance among different systems. Very specific requirements must be quantified and taken into account when doing capacity planning and deciding what machine configuration and tuning will best meet a customer's needs.

Conclusion

The announcement of the p690 16-way SPECsfs97_R1 results was a significant milestone for the IBM eServer pSeries brand, allowing IBM to claim industry leadership on the SPEC SFS 3.0 benchmark with this flagship datacenter-class UNIX server. And, as an exceptional server at the top of the eServer pSeries midrange family, the p660-6M1 8-way complements the high-end p690. Based on the published SPEC SFS 3.0 results, both of these machines are excellent choices for customers interested in a medium to large scale NFS server with outstanding fileserving performance characteristics.

References

[1] *Network Filesystem Specification*, Sun Microsystems, Inc., RFC-1094, (ftp://ftp.uu.net/inet/rfc/rfc1094.Z).

[2] *The NFS Version 3 Protocol Specification*, Sun Microsystems, Inc., RFC-1813, (ftp://ftp.uu.net/inet/rfc/rfc1813.Z).

[3] SPEC SFS97_R1 Results, February 5, 2002, (http://www.spec.org/osg/sfs97r1/results).

[4] SFS 3.0 Documentation Version 1.0, Standard Performance Evaluation Corporation (SPEC), December 1997.

[5] SPEC SFS97_R1 Press Release, Standard Performance Evaluation Corporation, September 26, 2001, (http://www.spec.org/osg/sfs97r1/press_release.html).

[6] *NFS V3 Design and Implementation*, Brian Pawlowski et. al., presented June 9, 1994: USENIX Summer 1994 - Boston, Massachusetts, (http://www.netapp.com/technology/level3/hitz94.html)

[7] *IBM eServer pSeries 690, Configuring for Performance*, October 2001, (http://www-1.ibm.com/servers/eserver/pseries/hardware/whitepapers/p690_config.html).

[8] *IBM eServer pSeries 660 Model 6M1, Technical Overview and Introduction*, Stephen Lutz and Shyan Manohar, September 4, 2001, Page 9, (http://www-1.ibm.com/servers/eserver/pseries/hardware/whitepapers/660_6m1_tech.pdf).

[9] *AIX 5L Version 5.1 Performance Management Guide*, Monitoring and Tuning File Systems section, Third Edition, December 2001, (http://publibn.boulder.ibm.com/doc_link/en_US/a_doc_lib/aixbman/prtfungd/prftungdtfrm.htm)

[10] SPEC SFS97 Results, June 14, 2001, (http://www.spec.org/osg/sfs97/results).

Special Notices

This document was produced in the United States. IBM may not offer the products, programs, services or features discussed herein in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the products, programs, services, and features available in your area. Any reference to an IBM product, program, service or feature is not intended to state or imply that only IBM's product, program, service or feature may be used. Any functionally equivalent product, program, service or feature that does not infringe on any of IBM's intellectual property rights may be used instead of the IBM product, program, service or feature. IBM makes no representation or warranty regarding third-party products or services.

Information in this document concerning non-IBM products was obtained from the suppliers of these products, published announcement material or other publicly available sources. Sources for non-IBM list prices and performance numbers are taken from publicly available information including D.H. Brown, vendor announcements, vendor WWW Home Pages, SPEC Home Page, GPC (Graphics Processing Council) Home Page and TPC (Transaction Processing Performance Council) Home Page. IBM has not tested these products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. Send license inquires, in writing, to IBM Director of Licensing, IBM Corporation, New Castle Drive, Armonk, NY 10504-1785 USA.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. Contact your local IBM office or IBM authorized reseller for the full text of a specific Statement of General Direction.

The information contained in this document has not been submitted to any formal IBM test and is distributed "AS IS". While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. The use of this information or the implementation of any techniques described herein is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. Customers attempting to adapt these techniques to their own environments do so at their own risk.

IBM is not responsible for printing errors in this publication that result in pricing or information inaccuracies.

The information contained in this document represents the current views of IBM on the issues discussed as of the date of publication. IBM cannot guarantee the accuracy of any information presented after the date of publication.

All prices shown are IBM's suggested list prices; dealer prices may vary.

IBM products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

Information provided in this document and information contained on IBM's past and present Year 2000 Internet Web site pages regarding products and services offered by IBM and its subsidiaries are "Year 2000 Readiness Disclosures" under the Year 2000 Information and Readiness Disclosure Act of 1998, a U.S. statute enacted on October 19, 1998. IBM's Year 2000 Internet Web site pages have been and will continue to be our primary mechanism for communicating year 2000 information. Please see the "legal" icon on IBM's Year 2000 Web site (http://www.ibm.com/year2000) for further information regarding this statute and its applicability to IBM.

Any performance data contained in this document was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally-available systems. Some measurements quoted in this document may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Trademarks

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM, AIX, e(logo), pSeries

IBM Trademarks information can be found at: http://www.ibm.com/legal/copytrade.shtml.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

SPEC, SPECjbb, SPECint, SPECfp, SPECweb, and SPECsfs are trademarks of the Standard Performance Evaluation Corporation and information can be found at: http://www.spec.org.

TPC, TPC-R, TPC-H, TPC-C, and TPC-W are trademarks of the Transaction Processing Performance Council. Information can be found at: http://www.tpc.org.

Other trademarks are the property of their respective owners.

Notes on Benchmarks and Values

The benchmarks and values shown here were derived using particular, well configured, development-level computer systems. Unless otherwise indicated for a system, the values were derived using 32-bit applications and external cache, if external cache is supported on the system. All benchmark values are provided "AS IS" and no warranties or guarantees are expressed or implied by IBM. Actual system performance may vary and is dependent upon many factors including system hardware configuration and software design and configuration. Buyers should consult other sources of information to evaluate the performance of systems they are considering buying and should consider conducting application oriented testing. For additional information about the benchmarks, values and systems tested, contact your local IBM office or IBM authorized reseller or access the following on the Web:

TPC	http://www.tpc.org
GPC	http://www.spec.org/gpc
SPEC	http://www.spec.org
Pro/E	http://www.proe.com
Linpack	http://www.netlib.no/netlib/benchmark/performance.ps
Notesbench Mail	http://www.notesbench.org
VolanoMark	http://www.volano.com

Unless otherwise indicated for a system, the performance benchmarks were conducted using AIX V4.2.1 or 4.3, IBM C Set++ for AIX/6000 V4.1.0.1, and AIX XL FORTRAN V5.1.0.0 with optimization where the compilers were used in the benchmark tests. The preprocessors used in the benchmark tests include KAP 3.2 for FORTRAN and KAP/C 1.4.2 from Kuck & Associates and VAST-2 v4.01X8 from Pacific-Sierra Research. The preprocessors were purchased separately from these vendors.

The following SPEC and Linpack benchmarks reflect the performance of the microprocessor, memory architecture, and compiler of the tested system:

- SPECint95 - SPEC component-level benchmark that measures integer performance. Result is the geometric mean of eight tests that comprise the CINT95 benchmark suite. All of these are written in the C language. SPECint_base95 is the result of the same tests as CINT95 with a maximum of four compiler flags that must be used in all eight tests.

- SPECint_rate95 - Geometric average of the eight SPEC rates from the SPEC integer tests (CINT95). SPECint_base_rate95 is the result of the same tests as CINT95 with a maximum of four compiler flags that must be used in all eight tests.

- SPECfp95 - SPEC component-level benchmark that measures floating-point performance. Result is the geometric mean of ten tests, all written in FORTRAN, that are included in the CFP95 benchmark suite. SPECfp_base95 is the result of the same tests as CFP95 with a maximum of four compiler flags that must be used in all ten tests.

- SPECfp_rate95 - Geometric average of the ten SPEC rates from SPEC floating-point tests (CFP95). SPECfp_base_rate95 is the result of the same tests as CFP95 with a maximum of four compiler flags that must be used in all ten tests.

- SPECint2000 - New SPEC component-level benchmark that measures integer performance. Result is the geometric mean of twelve tests that comprise the CINT2000 benchmark suite. All of these are written in C language except for one which is in C++. SPECint_base2000 is the result of the same tests as CINT2000 with a maximum of four compiler options that must be used in all twelve tests.

- SPECint_rate2000 - Geometric average of the twelve SPEC rates from the SPEC integer tests (CINT2000). SPECint_base_rate2000 is the result of the same tests as CINT2000 with a maximum of four compiler options that must be used in all twelve tests.

- SPECfp2000 - New SPEC component-level benchmark that measures floating-point performance. Result is the geometric mean of fourteen tests, all written in FORTRAN and C languages, that are included in the CFP2000 benchmark suite. SPECfp_base2000 is the result of the same tests as CFP2000 with a maximum of four compiler options that must be used in all fourteen tests.

- SPECfp_rate2000 - Geometric average of the fourteen SPEC rates from SPEC floating-point tests (CFP2000). SPEC_base_rate2000 is the result of the same tests as CFP2000 with a maximum of four compiler options that must be used in all fourteen tests.

- SPECweb96 - Maximum number of Hypertext Transfer Protocol (HTTP) operations per second achieved on the SPECweb96 benchmark without significant degradation of response time. The Web server software is ZEUS v.1.1 from Zeus Technology Ltd.

- SPECweb99 - Number of conforming, simultaneous connections the Web server can support using a predefined workload. The SPECweb99 test harnass emulates clients sending the HTTP requests in the workload over slow Internet connections to the Web server. The Web server softwre is Zeus from Zeus Technology Ltd.

- LINPACK DP (Double Precision) - n=100 is the array size. The results are measured in megaflops (MFLOPS).

- LINPACK SP (Single Precision) - n=100 is the array size. The results are measured in MFLOPS.

- LINPACK TPP (Toward Peak Performance) - n=1,000 is the array size. The results are measured in MFLOPS.

- LINPACK HPC (Highly Parallel Computing) - solve largest system of linear equations possible. The results are measured in GFLOPS.

VolanoMark is a 100% Pure Java™ server benchmark characterized by long-lasting network connections and high thread counts. In this context, long-lasting means the connections last several minutes or longer, rather than just a few seconds. The VolanoMark benchmark creates client connections in groups of 20 and measures how long it takes for the clients to take turns broadcasting their messages to the group. At the end of the test, it reports a score as the average number of messages transferred by the server per second.

VolanoMark 2.1.2 local performance test measures throughput in messages per second. The final score is the average of the best two out of three results.

The following SPEC benchmark reflects the performance of the microprocessor, memory subsystem, disk subsystem, network subsystem:

- SPECsfs97_R1 - the SPECsfs97_R1 (or SPEC SFS 3.0) benchmark consists of two separate workloads, one for NFS V2 and one for NFS V3, which report two distinct metrics, SPECsfs97_R1.v2 and SPECsfs97_R1.v3, respectively. The metrics consist of a throughput component and an overall response time measure. The throughput (measured in operations per second) is the primary component used when comparing SFS performance between systems. The overall response time (average response time per operation) is a measure of how quickly the server responds to NFS operation requests over the range of tested throughput loads.

The following Transaction Processing Performance Council (TPC) benchmarks reflect the performance of the microprocessor, memory subsystem, disk subsystem, and some portions of the network:

- tpmC - TPC Benchmark C throughput measured as the average number of transactions processed per minute during a valid TPC-C configuration run of at least twenty minutes.

- $/tpmC - TPC Benchmark C price/performance ratio reflects the estimated five year total cost of ownership for system hardware, software, and maintenance and is determined by dividing such estimated total cost by the tpmC for the system.

- QppH is the power metric of TPC-H and is based on a geometric mean of the 17 TPC-H queries, the insert test, and the delete test. It measures the ability of the system to give a single user the best possible response time by harnessing all available resources. QppH is scaled based on database size from 30GB to 1TB.

- QthH is the throughput metric of TPC-H and is a classical throughput measurement characterizing the ability of the system to support a multiuser workload in a balanced way. A number of query users is chosen, each of which must execute the full set of 17 queries in a different order. In the background, there is an update stream running a series of insert/delete operations. QthH is scaled based on the database size from 30GB to 1TB.

- $/QphH is the price/performance metric for the TPC-H benchmark where QphD is the geometric mean of QppH and QthH. The price is the five-year cost of ownership for the tested configuration and includes maintenance and software support.

The following graphics benchmarks reflect the performance of the microprocessor, memory subsystem, and graphics adapter:

- SPECxpc results - Xmark93 is the weighted geometric mean of 447 tests executed in the x11perf suite and is an indicator of 2D graphics performance in an X environment. Larger values indicate better performance.

- SPECplb results (graPHIGS) - PLBwire93 and PLBsurf93 are geometric means of literal and optimized Picture Level Benchmark (PLB) tests for 3D wireframe and 3D surface tests, respectively. The benchmark and tests were developed by the Graphics Performance Characterization (GPC) Committee. The results shown used the graPHIGS API. Larger values indicate better performance.

- SPECopc results - CDRS-03, CDRS-04, DX-03, DX-04, DX-05, DRV-04, DRV-05, DRV-06, Light-01, Light-02, Light-02, AWadvs-01, AWadvs-02, Awadvs-03, and ProCDRS-02 are weighted geometric means of individual viewset metrics. The viewsets were developed by ISVs (independent software vendors) with the assistance of OPC (OpenGL Performance Characterization) member companies. Larger values indicate better performance.

The following graphics benchmarks reflect the performance of the microprocessor, memory subsystem, graphics adapter, and disk subsystem:

Bench95 and Bench97 Pro/E results - Bench95 and Bench97 Pro/E benchmarks have been developed by Texas Instruments to measure UNIX and Windows NT® workstations in a comparable real-world environment. Results shown are in minutes. Lower numbers indicate better performance.

The NotesBench Mail workload simulates users reading and sending mail. A simulated user will execute a prescribed set of functions 4 times per hour and will generate mail traffic about every 90 minutes. Performance metrics are:

- NotesMark - transactions/minute (TPM).
- NotesBench users - number of client (user) sessions being simulated by the NotesBench workload.
- $/NotesMark - ratio of total system cost divided by the NotesMark (TPM) achieved on the Mail workload.
- $/User - ratio of total system cost divided by the number of client sessions successfully simulated for the Mail NotesBench workload measured.

Total system cost is the price of the server under test to the customer, including hardware, operating system, and Domino Server licenses.