Meeting the Storage Challenges of Today's Linux® Grid Computing Enterprise

Network Appliance, Inc. | 6/3/04 | TR-3325

TECHNICAL REPORT

Network Appliance, a pioneer and industry leader in data storage technology, helps organizations understand and meet complex technical challenges with advanced storage solutions and global data management strategies.

ABSTRACT

Today's data-intensive applications are placing enormous pressures on Linux compute grids and their respective back-end storage systems. Whether it is rendering the latest animated scene for a new movie blockbuster, creating the latest automotive innovation, or reaching beneath the earth to find new energy reserves, compute grids are being put to the test day in and day out.

However, storage bottlenecks due to unbalanced workloads can drastically impact system performance and availability. Compute grids therefore need high-performance storage solutions that are flexible enough to handle the changing data demands based on shifting priorities and expanding workloads. Compute grids need storage solutions that can facilitate access to data related to high-priority jobs without disruption. And they need storage solutions that can scale and rebalance data without downtime.

This paper discusses the use of grid computing in various industry segments, reviews the challenges faced by these industries, presents an in-depth technical discussion of the NetApp SpinServer® solution, and reviews SpinServer performance tests conducted by an independent lab.

Table of Contents

1) Introduction to Grid Computing

Until recently, the concept of a high-performance computing (HPC) grid was of interest mainly to the scientific community. It was an idea rarely embraced by the corporate enterprise. Today, however, corporate computing grids have emerged as a cost-effective way to achieve the goals of HPC. In many cases replacing the need for expensive, proprietary supercomputers, grid computing deployment today allows the enterprise to achieve HPC while leveraging existing IT infrastructure, improving business processes, lowering costs, and speeding time-to-market.

As illustrated in Figure 1, a typical computing grid consists of a large number of high-performance servers—often referred to as a compute farm or a server farm—linked to large community of users by a high-speed network and sharing storage on the back end. While a compute grid typically resides within one location or department, an enterprise can contain multiple grids. With higher performance at lower costs being the primary business driver for grid computing, inexpensive, off-the-shelf Linux blade servers have emerged as the servers of choice for these environments. A single grid computing deployment today can easily contain thousands of Linux servers.

While grid computing can be found in just about every industry segment, this paper focuses on the following industries, where grid computing is particularly prevalent:

- Oil and gas exploration (for seismic data processing and reservoir simulation)
- Media and entertainment (for digital movie rendering, editing, special effects, and animation)
- Auto manufacturing (for design, analysis, simulation, and manufacturing)
- Semiconductor manufacturing (for chip design, simulation, verification, and layout)
- Software development (for compilation and testing)

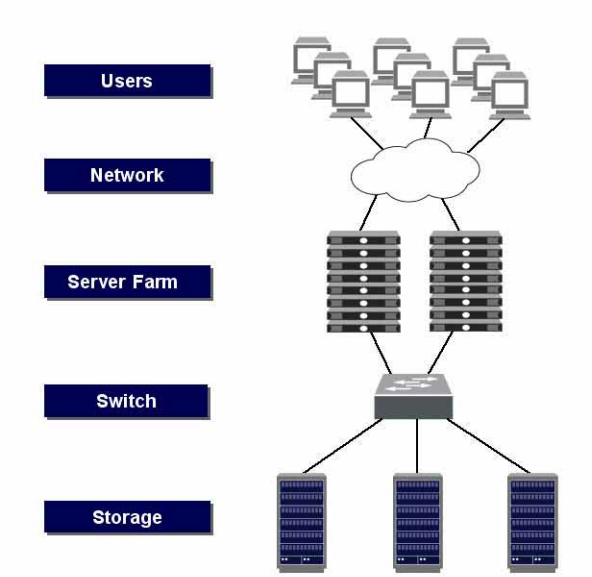

Figure 1) Typical Linux-based grid computing environment.

2) Grid Storage Challenges

While compute grids have become a critical, central resource for many companies, these environments are facing complex challenges for back-end data storage. Multiple mission-critical projects are running jobs on a compute grid at any given time—projects such as automotive design and development, animation for a movie, or software code development. The data storage needs for such projects are typically very large—up to hundreds of terabytes—complicated by the fact that hundreds or thousands of users need high-performance, concurrent access to this data.

2.1 Shifting Priorities

As in any business or industry, the priority associated with each of these projects may change—not only once but usually several times.

For example, a given automobile assembly may suddenly be needed earlier than anticipated because of a design change. The release date for a movie is moved up, delayed, and then unexpectedly moved up again to meet a last-minute marketing or distribution decision. A chip design needs to “tape out” earlier than expected. Within the seismic processing world, a client may request a new analysis on a given piece of property, wanting the analysis to have top priority—a service for which the client is willing to pay. The company will have to modify current plans and dynamically reallocate resources to accommodate the turnaround requested by the client.

As the priority of a new or existing project changes or shifts back and forth dynamically, so does the demand for the large amounts of data related to that project. To top it all off, this can be happening with multiple projects simultaneously.

2.2 Expanding Grids

Because Linux computers are relatively inexpensive, companies can readily purchase additional compute servers to expand their aggregate processing power on the front end—so that they can take on more projects and thereby increase revenues. This in turn increases the amount of data being stored on the back end—not to mention the demands it places on the storage system that is serving up the mission-critical data.

2.3 Overburdened, Unbalanced Storage Systems

Changing and increased demands for data based on shifting priorities and increased workloads can overburden some storage systems as jobs “back up" for processing. Consider the following typical scenario: a storage device containing data related to a top-priority or “hot” project is constantly being accessed and locking up—creating storage “hot spots.” Meanwhile another storage device containing lesser-priority data is being underutilized.

This unbalanced storage situation is illustrated in Figure 2, where 50% to 80% of the calls are going to 20% of the storage devices. Figure 3, in contrast, illustrates a balanced storage environment where the load is evenly distributed.

When storage becomes overburdened, the IT department usually cannot add storage devices quickly enough to accommodate and manage the additional data being generated by more projects and newly added Linux nodes on the front end. Even when IT is able to add more physical storage, there is often a migration period in which the “hot data” is split into smaller portions and distributed onto the new namespace. During this migration, the cluster nodes acting upon the data are dormant.

Dynamic and growing demands not only impact the operations staff, but also have a negative impact on performance. New projects require new I/O cycles, thereby requiring more throughput to satisfy the compute cycles added to the cluster. As hot spots develop, operations are impacted, and users complain about turnaround times and schedule delays.

The bottom line is that storage hot spots and bottlenecks are being created every day because of unbalanced workloads, which can drastically impact overall system performance and availability. Computing grid environments therefore need high-performance storage grids that are flexible enough to handle the changing demands for data—based on shifting priorities, expanding workloads, and growing user data access requirements. Compute grids require storage solutions that can facilitate access to data related to high-priority jobs without user disruption. They require flexible storage solutions that can scale and rebalance data without downtime and with little or no impact on operations.

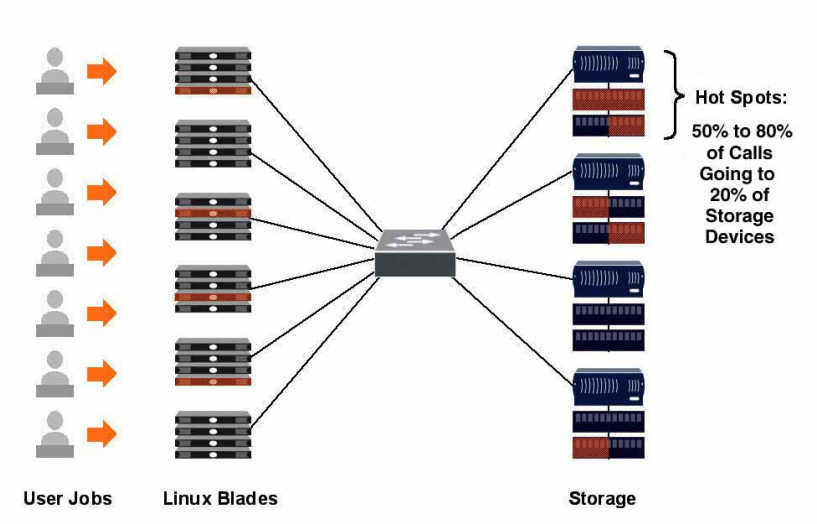

Figure 2) Storage hot spots in a grid computing environment.

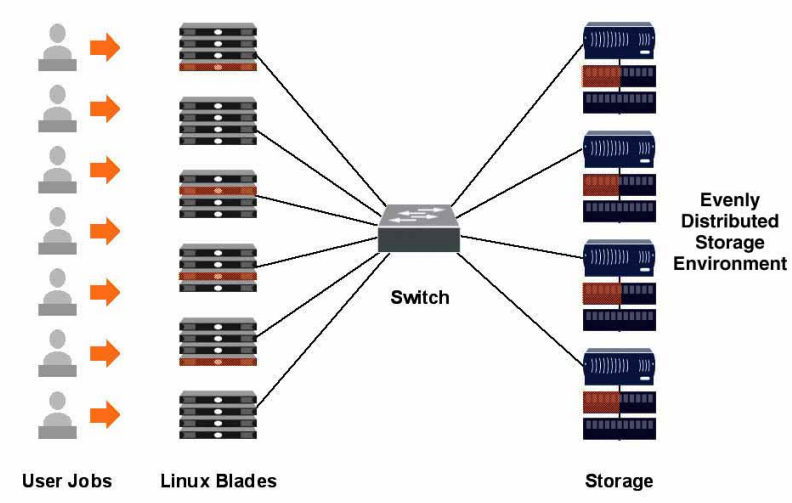

Figure 3) An evenly distributed grid environment without storage hot spots.

3) Challenges Faced by Specific Industries

Regardless of the industry, today's data-intensive applications are placing enormous pressures on compute grids, back-end storage systems, and the operations personnel responsible for them. Table 1 summarizes the typical applications and challenges faced by the major industries increasingly committed to grid computing solutions.

Table 1) Grid computing applications and challenges by industry.

Industry	Applications	Data Sets	Challenges
Oil and Gas	Seismic processing	Very large image data sets; many intermediate versions	Very high aggregate I/O to storage; multiple jobs generate hot spots
	Reservoir modeling	Many small files	Massively compute bound
Entertainment	Renderman; Maya, Softimage; ray tracing	Very large files; 2D and 3D frames; textures; compositing	Hot spots; concurrent access to data sets
Automotive	Computational fluid dynamics; crash simulation; finite element analysis	Large files	Data availability; performance; storage hot spots
Semiconductors	Cadence Verilog; Synopsys VCS	Large files	Scalability; availability; performance; hot spots
Software Development	Rational ClearCase	Mixed, large, and small files; replicated source trees	Remote collaboration; hot spots during compilation

3.1 Oil and Gas Exploration

3.1.1 The Industry

Today's HPC advances are revolutionizing the oil and gas industry. Increasing the accuracy and quality of the subsurface picture greatly enhances the likelihood of drilling successes. New energy-rich locations can quickly be identified, poor prospects can be eliminated, and existing reservoirs can be more fully recovered.

This new power can be directly traced to the introduction of "commodity" 64-bit processor technology and the emergence of high-performance grid computing—thereby allowing the use of wave equation migration algorithms. Seismic image processing has always been an essential complement to the data acquisition process. In the past, limitations in technology and prohibitive costs were significant barriers to large-scale usage and the application of more advanced algorithms. With the advent of inexpensive Linux computing, these barriers have been reduced. Fast, high-quality seismic processing is now available to oil and gas companies of all sizes.

Geophysicists now have increased access to higher-fidelity computational models to generate more precise analysis of the earth's subsurface. In addition, advanced software tools that simulate the migration of oil and gas through sedimentary layers as an underground reservoir is extracted can reduce the costs of oil extraction and production. Running on Linux-based compute grids for days or even weeks at a time, these comprehensive seismic models and simulations generate the 3D subsurface images that increasingly guide drilling and extraction operations.

3.1.2 Storage Challenges

These advanced computational techniques in turn place increased demands on the back-end storage systems. The data requirements generated by these seismic applications are enormous. Cycle time delays in the exploration and production process can impose millions of dollars of additional costs to the oil and gas discovery process.

Large amounts of data need to be shared by different groups on a truly global scale. Seismic processing and reservoir simulation output data must be analyzed and interpreted by teams of geoscientists and engineers, sometimes on opposite ends of the world. Some of the 3D immersive visualization applications require very fast, burst access to seismic cube volumes. Remote collaboration between teams—on the same floor or on different continents—imposes critical new challenges.

In addition, the sheer size of the data used in this industry creates very large data management issues. The millions of tapes and petabytes of disk space contain data that has value only if it can be inventoried, archived, and recovered for use in follow-on analysis as an oil or gas field is depleted over decades of extraction.

3.2 Media and Entertainment

3.2.1 The Industry

The media and entertainment industries have emerged as major forces in the advancement of grid computing technology. Nowhere is this more evident than in the specialized field of digital animation, rendering, and special effects generation.

Over the past decade, animation and special effects designers have relied on HPC solutions in the form of large-scale UNIX® multiprocessor systems or proprietary supercomputers with direct-attached disk arrays to render the complex images for movie blockbusters. More recently, industry-leading special effects organizations have begun moving away from high-end multiprocessor systems to Linux-based compute farms with off-the-shelf hardware.

3.2.2 Storage Challenges

As entertainment industry firms move to grid computing environments, however, they are quickly finding that this new computing model can rapidly overwhelm their traditional file server implementations. Consider the example of rendering a special effects sequence. In a typical sequence, each frame tends to need the same data. Using the traditional, centralized multiprocessor approach, the first frame would pick up that data, and the succeeding frames would simply read it out of cache. With a compute grid approach, however, things are not so simple. Instead of serving up data to just 50 or 60 multiprocessors, the file servers must now feed data to thousands of servers.

This can cause network and disk I/O traffic to “go through the roof.” During busy production times, it would not take long to literally reach a file server meltdown, and production would come to a grinding halt. If 1,000 render nodes all need to apply the same texture data to a sequence of frames, the server that contains the texture information is doomed. During a deadline crunch, it is not practical to try to install additional file servers to meet the exploding storage demand. A better storage solution is needed.

The ability to seamlessly move, replicate, and distribute hot data behind a cluster of storage servers is the “holy grail” for rendering houses.

3.3 Auto Manufacturing

3.3.1 The Industry

The auto industry is a driving force behind the grid computing phenomenon. Deploying very large Linux compute farms to process advanced applications such as computational fluid dynamics (CFD), automobile and automotive component manufacturers rely on computerized analyses and simulations to test design ideas before beginning production. By using high-speed computer simulation techniques to understand fluid flow and heat transfer, automotive design engineers can predict the success of new products before undertaking the production of time-consuming and expensive prototypes.

In order to run CFD applications, high-performance hardware platforms such as Linux compute farms are typically deployed. The net result is a faster design and analysis process, which in turn can lead to shorter production cycles, faster time-to-market, and lower costs.

3.3.2 Storage Challenges

Data storage challenges can often forestall the aforementioned efficiencies and savings, however. For example, design and engineering teams require seamless system accessibility and interoperability across Windows®, UNIX, and Linux platforms. For these companies, the ability to dependably share and access information is vital to the product design, development, and manufacturing process.

In addition, data tends to grow exponentially with each successive step in the design and development process. As data continues to grow, so does the overall risk associated with the potential loss of that data. Data sets and files associated with the design and analysis stages are especially mission-critical. The level and quality of data management have a direct effect on engineering performance and productivity—and ultimately on the company's time-to-market.

Design collaboration is an additional challenge in the auto industry. Major automotive manufacturers may employ thousands of mechanical engineers, designers, support staff, independent contractors, consultants, and business partners. Teams must be able to communicate and work efficiently and economically, regardless of where the individual members—and the data—reside physically.

3.4 Semiconductor Manufacturing

3.4.1 The Industry

Semiconductor design and manufacturing require access to extremely large amounts of data, especially during the design simulation process. It is not uncommon for a chip manufacturer to deploy several thousand Linux servers on a compute grid—each with one or more copies of simulation software such as Verilog from Cadence and VCS from Synopsys—to achieve the desired performance levels.

Often, resources tend to be used inefficiently, loads can be unevenly distributed, and quality of service can be compromised. In a fast-moving design environment not necessarily known to be capacity planning friendly, manufacturers need a fluid, elastic, and highly performing storage environment. Engineers and administrators must be able to make on-the-fly capacity allocations and storage modifications.

3.4.2 Storage Challenges

A single chip design can require as much as 10TB of storage. And many semiconductor manufacturers have multiple chip designs going through the design process every month. Most of this storage is usually for scratch space, while 2TB to 3TB of design data often needs to be retained in long-term storage. As a result, the IT department must forecast and manage extremely large amounts of data, which can be very difficult.

A fundamental challenge in this industry involves the need for the utmost in data availability. Should a company lose access to design-critical data, the preestablished “tape out” date may be missed—resulting in millions of dollars in lost profits. Data availability and management are further complicated by the remote and heterogeneous nature of the semiconductor design and manufacturing environment. As a result, comprehensive data protection is imperative.

3.4.3 A Typical Example

A leading semiconductor manufacturer has deployed one of the largest compute grids in the world, supporting over 3,500 Linux processors. The company’s chip designers send highly complex jobs to the servers—jobs that can take up to 10 days to run. If a server or its data storage device crashes near the end of that period, and the job has to be rerun, the setback can delay the company’s chip manufacturing process by as much as two weeks. Such a delay puts a product behind schedule, thereby damaging chances of meeting critical market share and revenue goals.

Previously, the company relied on direct-attached storage. But as demand for the company's chips skyrocketed, so did the need for more storage space. To handle the increasing workload, the company rapidly added hardware engineers, each of whom could need as much as 40GB of scratch space on the storage system. The servers used by the engineering department quickly became overloaded, because they had to process highly CPU-intensive jobs and handle I/O processes for the NFS files stored locally.

Anticipating a continuing increase in the demand placed on its servers and storage systems, the company reevaluated its overall data storage approach and selected a NetApp storage solution. The advantages to the company included 99.99^{+}% uptime, offloading of I/O operations from servers, rapid scalability, and virtually no administration overhead.

3.5 Software Development

3.5.1 The Industry

Over the past few years, software development has become one of the more challenging fields in the technology industry. Whereas software developers once had many months and sometimes years to create each new release, development cycles are now continuously and significantly compressed. Customer expectations are that each new release must be more powerful and feature-rich than its predecessor, be of higher quality, and support multiple platforms.

Add to that the fact that software development firms are now routinely deploying sophisticated software configuration management (SCM) platforms such as IBM Rational ClearCase to aid in the development and management of highly evolved software products containing millions of lines of code, and the result can be an extremely complex, power-hungry development environment.

3.5.2 Storage Challenges

The major business issue faced by software development organizations is how to deliver the finished software product more quickly and at a lower cost than the competition. To meet this business issue, development firms are looking for ways to cost-effectively bolster overall processing power, increase staff productivity, heighten operational efficiencies, streamline resources, compress development and testing times, and reduce overall operating costs. These are just some of the reasons why software development organizations are increasingly deploying high-performance, flexible, cost-effective grid computing solutions.

From a storage standpoint, software teams working in grid computing environments face a number of challenges, particularly during the very critical compilation and testing phases of the development process. These challenges include storage load balancing to prevent hot spots, storage scalability, data migration, and remote collaboration. In addition, ease of use and ease of management are important issues for this market segment.

4) Evaluating Available Storage Grid Solutions

There are a number of storage solutions available on the market today that address the data storage challenges faced by the Linux grid computing enterprise. When reviewing alternative storage solutions, it is recommended that the following areas be given careful attention.

4.1 Performance

One of the biggest issues in building an enterprise-level compute grid with thousands of Linux servers is finding a back-end storage solution that provides comparable aggregate bandwidth. Some storage solutions are tuned for streaming I/O, while others target high-NFS operations. The best-of-class solution will have the ability to address both needs. Throughput is directly related to the number of network connections available to serve up the namespace. Having a petabyte of storage behind 4GB interfaces will not make the process go any more quickly.

4.2 Scalability

It is important that the back-end data storage solution have the ability to scale to the future needs of the enterprise. The solution should provide the ability to add additional file servers easily and transparently within the same file system image or namespace—without disrupting users. Some available storage solutions have an upper limit of only 6TB, while others can scale to as much as11,000TB—an important distinction. If the storage growth capacity is there, adding additional disk space can be a relatively simple process. However, the real challenge involves migrating and moving the existing data to the new disk space.

4.3 Transparency

The ability to add performance and capacity independently, migrate data, and change data priorities without impacting user productivity is paramount. Not all server solutions provide these capabilities.

4.4 Load Balancing

Another issue with grid computing is determining how to distribute the data to eliminate hot spots and maximize the performance of the most critical jobs. The data storage solution should provide load-balancing facilities that make it easy to accomplish such distribution without disrupting access to the storage system. Load-balancing functionality is not available with all storage solutions on the market today.

4.5 Snapshot™ Copies

Snapshot copies are important tools for a variety of reasons—including quick recovery of deleted or corrupted data. Some available storage solutions support only 32 Snapshot copies per volume. Others can support thousands of Snapshot copies per virtual file system. In addition, it is important to ensure that the Snapshot process does not adversely affect the file-serving activity.

4.6 Asynchronous Mirroring

For data that changes relatively rarely, the availability of asynchronous mirroring to distribute multiple copies of data throughout the storage cluster can be a valuable feature. Keep in mind, however, that mirroring in and of itself does not buy anything unless it can be accessed from a different set of interfaces than the original source data.

4.7 Ease of Management and Use

Finally, it is important to consider ease-of-management and ease-of-use issues. For example, the ability to manage a shared storage infrastructure where objects can be viewed, managed, and prioritized from a single management window with all clustered resources available can be a critical capability. Having the ability to write customized scripts to create and manage storage resources is another task to consider when choosing a storage solution.

5) The NetApp SpinServer Solution

The Network Appliance™ SpinServer solution is a flexible, high-performance grid storage implementation that scales to hundreds of file servers within a single SpinServer cluster. Specifically, a SpinServer cluster can grow to encompass as many as 512 servers and a total of 11,000TB of storage. This scalability enables SpinServer to provide sufficient aggregate bandwidth to support a grid computing environment encompassing thousands of Linux compute servers.

Jamie Gruener, senior analyst at The Yankee Group, describes SpinServer as follows:

> Enterprises that have deployed large-scale, Linux-based grid computing environments are often impacted by bottlenecks caused by shifting job priorities and unbalanced workloads, as well as poor management tools. This negatively impacts overall system performance and availability. These enterprises are seeking flexible, grid-based storage environments that can handle the changing demands for data based on ever-shifting priorities. Products such as NetApp SpinServer can facilitate access to data related to 'hot' jobs without disruption, improve the management of data across multiple nodes, and scale and rebalance data without downtime.

SpinServer provides performance optimization via data migration. The solution transparently moves data from an overburdened or hot SpinServer cluster to an existing or newly added SpinServer cluster that is not as burdened. Additionally, SpinServer provides a global namespace for all data stored in the SpinServer cluster. The benefit of global namespaces is that the data movement is transparent to the end user, since the data remains in the same logical place in the global namespace before, during, and after the move.

5.1 Advanced Functionality

SpinServer provides a number of advanced functions and features that make it an excellent choice as a back-end storage solution for a Linux-based compute grid. For example, the aggregate throughput of a SpinServer cluster scales linearly with the number of servers exporting the global file system, enabling the SpinServer cluster to grow along with the compute cluster. The SpinServer cluster can be dynamically reconfigured without service outages and disruption to users. It provides online mechanisms for recovering from "disk full" conditions, modifying job priorities, and adding capacity in the form of new servers. In addition, data in a SpinServer cluster can be asynchronously mirrored to increase the achievable aggregate read throughput to that data.

5.2 SpinServer Architecture

The SpinServer architecture is based on the concept of a *cluster*, which is a collection of SpinServer systems connected through a *cluster interconnect*. The cluster interconnect is an IP network, typically implemented with a switched Gigabit Ethernet network.

Each SpinServer system contains one or more storage pools. A *storage pool* is a collection of Fibre Channel LUNs. Typically, the LUNs that make up a storage pool are of the same storage class—such as 10,000 RPM drives—that are striped or concatenated together to comprise a single array of disk blocks with very high aggregate performance.

Every storage pool, in turn, contains one or more virtual file systems (VFSs). VFSs are directory trees, each consisting of a topmost directory and a set of nested subdirectories and their associated files. Instead of carrying its own array of blocks, a VFS includes just an inode file, which points to the file, directory, and metadata blocks stored in the VFS. The data contents of the VFS can reside on any subset of blocks within the storage pool. This allows for easy transportability between the virtual and physical realms. As files are deleted from a VFS, their blocks are automatically freed back to the storage pool. As files are created or modified, any free block from the storage pool can be allocated to any VFS in the pool.

5.3 SpinServer Global Namespace

A SpinServer cluster exports a single global file system by “gluing together” a set of VFSs into a single tree. One VFS is identified as the *root VFS*, which defines the root directory of the global file system. Other VFSs are spliced into this global file system by creating special files—called *mount points*—which specify the VFS to be linked at that location in the global namespace.

These mount points are visible only internally within the SpinServer system. They are unrelated to normal UNIX or Linux mount points. From the perspective of an NFS client, there is a single NFS export, which represents the entire global file system that is exported by the SpinServer cluster. Behind the scenes, the global file system is actually implemented by the collection of VFSs within the SpinServer cluster.

The SpinServer architecture that supports this global namespace is illustrated in Figure 4. This figure shows a SpinServer cluster of four servers: **SS 1**, **SS 2**, **SS 3**, and **SS 4**. The VFSs—**eng**, **proj_abc**, **proj_xyx**, **hw**, **sw**, **des**, **sim**, and **syn**—are located on the various SpinServer systems, as shown on the left, and are organized into the corresponding global namespace, as shown on the right. This global namespace is the NFS file system as it appears to all NFS clients.

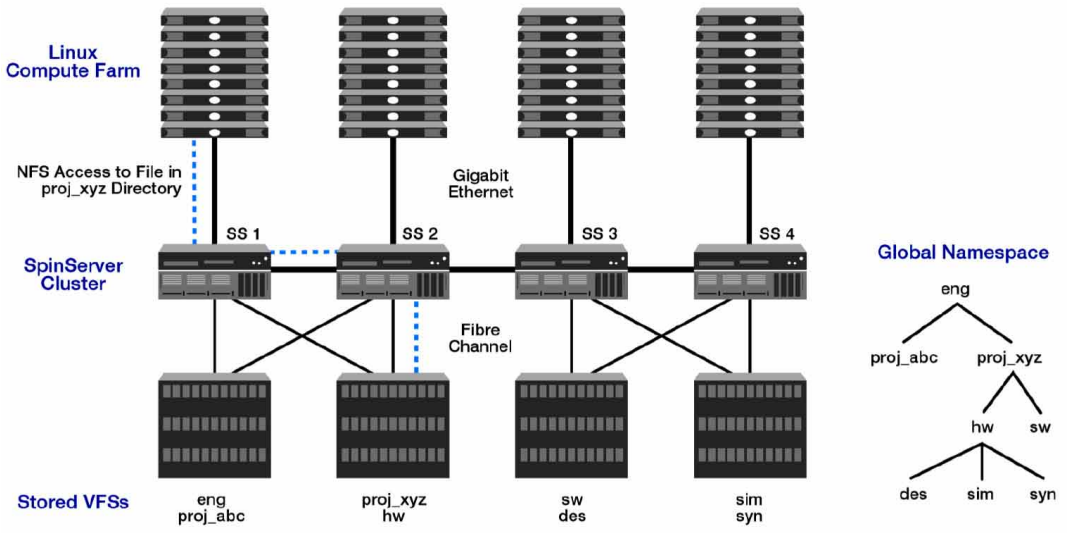

Figure 4) SpinServer architecture.

Figure 4 shows incoming NFS requests entering the cluster from the top. The lines connecting the SpinServer systems to each other are cluster Ethernet links, and the lines connecting SpinServer systems to their disks are Fibre Channel links.

In this figure, a member of the leftmost group of Linux clients is accessing a file in the **proj_xyz** directory, which is stored in a VFS on **SS 2** (refer to the dotted line). The operation begins at **SS 1**, which receives the request and, after determining that the referenced VFS is not local, forwards the request on to **SS 2** via the switched Ethernet cluster link. SS 2 receives the forwarded request, executes it, and then sends the response back to **SS 1**.

Architecturally, a SpinServer system can be thought of as an integrated NAS switch and server, where requests arrive at a SpinServer system and, if they refer to nonlocal data, are then switched to the server that is actually storing the referenced data. Since this is a switching architecture, the aggregate bandwidth that is available through the cluster scales linearly with the number of servers, enabling the creation of clusters that deliver very high bandwidth.

5.4 Virtual File System Movement

The location of a VFS in the global namespace is completely independent of the SpinServer system storing that VFS. NetApp makes use of this autonomy to provide dynamic online reconfiguration of the system by supporting online movement of VFSs between SpinServer systems.

A storage administrator can perform a “VFS Move” operation, which copies the contents of a VFS from one server to another, at any time. Read and write operations can still be performed against the VFS during the move operation, and these updates are propagated during the second stage of the move. The final stage of the move operation transfers the NFS file-locking state between the source and destination servers.

Thus, VFS Move can transfer a VFS from one server to another, while users are accessing the files in that VFS, while NFS users have files locked. The names and locations of the files in the global namespace do not change as a result of the move operation, nor do any mount points at the NFS clients become invalid.

6) Tailoring SpinServer for Linux Compute Grids

This section describes how to use a SpinServer cluster as back-end storage for a Linux compute grid. It discusses scaling overall performance, rebalancing resources to change overall priorities, rebalancing resources after adding a new server, handling exceptional events such as disk full errors, and using mirrors to increase read bandwidth to data.

6.1 Performance

The biggest issue in building a Linux compute grid with thousands of servers is finding a back-end storage solution that provides comparable aggregate bandwidth. The NetApp SpinServer cluster is well equipped to provide this level of performance to a single NFS exported file system.

As was shown in Figure 4, the key to scaling the overall performance for a SpinServer cluster is distributing the loaded VFSs relatively evenly among the servers in the cluster. With an even distribution, a SpinServer cluster can scale linearly with the number of SpinServer systems, up to a total of 512 systems. Typically, in a cluster that is load-balanced in this way, the majority of the traffic will be transmitted using the cluster network, which has been specifically designed to carry this load.

6.2 Rebalancing Loads

The greatest benefits in using a SpinServer cluster for the back-end storage for a Linux compute farm are realized when dealing with dynamically changing environments.

These benefits result from the combination of several features. The global namespace that is exported by a SpinServer cluster is completely independent of the physical location of the VFSs that comprise the global file system. In addition, a VFS can be moved atomically from one SpinServer system to another, while it is being accessed, without any visible changes to the storage clients. This combination of features enables a VFS to be relocated by a storage administrator at any time and without any user-visible changes.

A common problem with Linux compute farms is determining how to distribute the data to maximize the performance of the most critical jobs. This distribution can be accomplished easily and without disrupting access to any storage by using a SpinServer system. The VFSs that are used by the critical jobs can be distributed evenly among the fastest SpinServer systems, while the VFSs that are used by lower-priority jobs can be moved to a smaller set of servers or to servers with slower storage.

For example, in Figure 5, VFSs **A1**, **A2**, and **A3** store the data for job **A**, and the other, similarly named VFSs store the data for jobs **B**, **C**, and **D**. Any given job accesses storage on three of the servers, and whenever any two jobs are running, all of the servers are supplying data to the compute cluster.

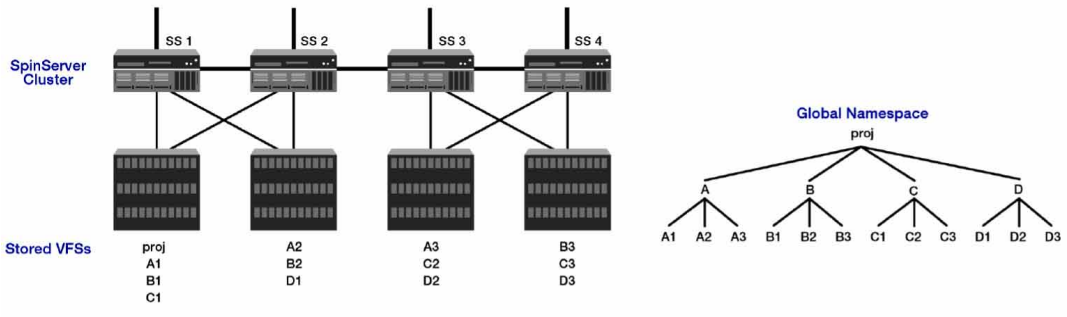

Figure 5) Even VFS distribution for equal job priority.

If jobs **A** and **D** become the highest priority, the administrator can reconfigure the storage as shown in Figure 6, so that the VFSs for the highest priority jobs are stored on the fastest drives and are backed by the most servers. The remaining jobs make use of the remaining resources. In this example, three servers in the cluster serve jobs **A** and **D**, while jobs **B** and **C** share a single server.

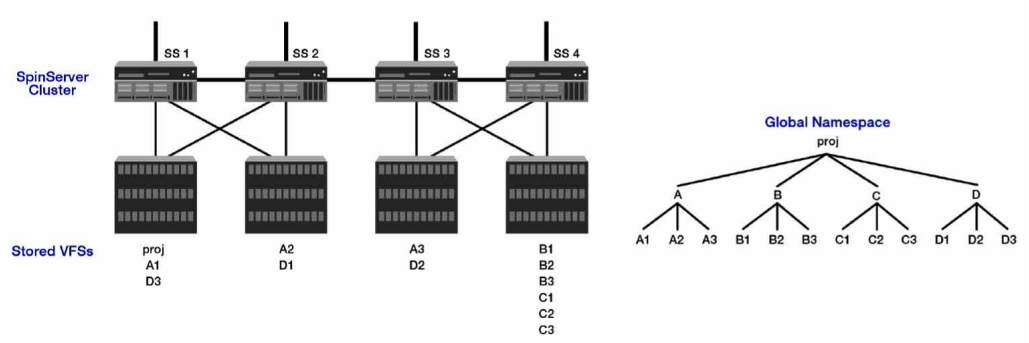

Figure 6) Prioritizing jobs A and D.

Load balancing is also performed after bringing a new server into production. In this case, the cluster starts with a new, empty server, as shown in Figure 7.

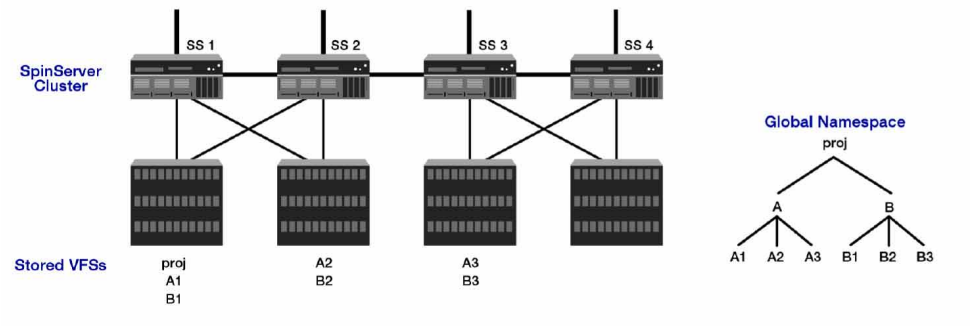

Figure 7) Adding a new server—before load balancing.

The administrator can then select certain VFSs to move to the new server, SS 4. If the two busiest servers are SS 1 and SS 2, the administrator might choose VFSs from those servers to move, as shown in Figure 8.

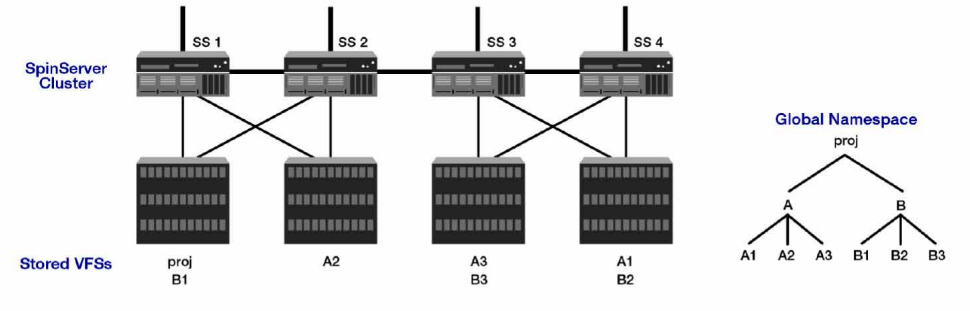

Figure 8) Adding a new server—after load balancing.

During and after the moves, the global namespace remains unchanged. Since the VFSs move atomically, this load balancing is a completely nondisruptive operation.

Contrast this with the behavior of traditional server clusters. For example, consider a scenario in which a system administrator has two servers, each storing 6TB of data, and wants to add a third server. To rebalance the storage so that it is evenly distributed among all three servers, 4TB of data should be copied to the new server. At 1GB per second, this transfer can take up to 9.5 hours to complete, during which time the relocated data is unavailable to users. After the move is complete, since the path name to the moved data has changed, the administrator must now update all of the scripts and programs that include the path names of the moved data. The administrator must also update every client in the cluster with the new file server name and mount point.

6.3 Disk Full Errors

The SpinServer architecture makes dealing with many exceptional events very simple. One of the most common exceptional events is running out of disk space. At a VFS level, increasing capacity is as simple as running a command to expand the size of the VFS, providing there is enough free space in the storage pool. Alternatively, the VFS can be moved to a less populated pool in the cluster, where its size can be increased once the move is complete. Another possible scenario is when all storage pools have run out of space. In this case, a storage pool can be added seamlessly, and selected VFSs can be moved into the new pool.

Consider the original set of jobs, shown in Figure 5. If the **B1** and **C1** VFSs begin to grow rapidly, the **SS 1** storage pool will eventually fill up. Before the disk is full, the administrator can redistribute the storage, without having to take the data offline. In this example, the administrator wants to redistribute the VFSs to ensure that **B1** and **C1** are on different servers, for example, by moving the **C1** VFS to **SS 2** and moving the D1 VFS to **SS 1**, as shown in Figure 9.

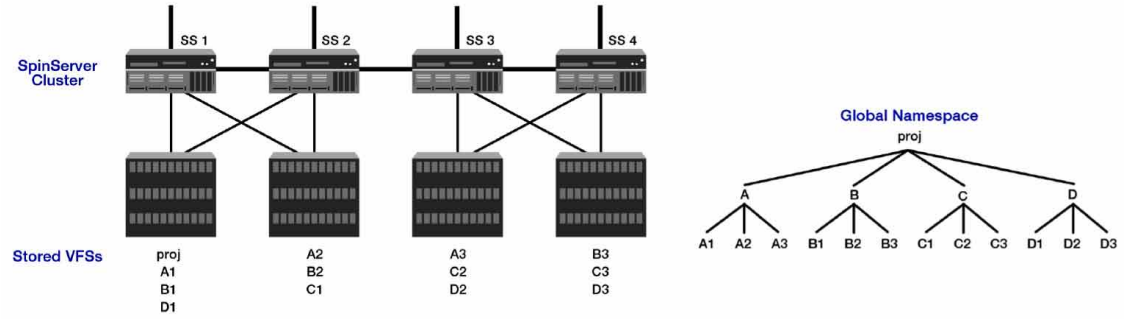

Figure 9) Avoiding disk full error by separating B1 and C1.

In Figure 9, the two rapidly growing VFSs, **B1** and **C1**, are now on separate SpinServer systems, but the global namespace remains completely unchanged. The “disk full” error condition has been remedied by increasing disk utilization instead of by purchasing additional storage. Furthermore, the problem was resolved without disrupting access to any storage from any client.

6.4 Asynchronous Mirroring

In some situations, an application requires more read bandwidth to a single file than a single SpinServer system can provide. For data that changes relatively rarely, an application can use *asynchronous mirroring* to distribute multiple copies of that data throughout the cluster.

Returning to the example in Figure 5, if there is high demand for the data in the **A3** VFS, **A3** can be mirrored across several SpinServer systems, resulting in the configuration shown in Figure 10. In this figure, the **A3** VFS is mirrored to **SS 2**, **SS 3**, and **SS 4**. As requests for **A3** are received by any port in the cluster, the requests are distributed in a round-robin fashion among the servers that are storing the mirrored **A3** VFS, effectively tripling the effective bandwidth to data stored in **A3**.

Since they are asynchronous and read-only, SpinServer mirrors are not suitable for all applications. To make effective use of SpinServer mirrors, an application must be prepared to see data that is at least 30 seconds out of date from the most recent version.

However, for suitable applications, mirroring can be a powerful tool. The mirrored data resides in a single location in the namespace, independent of the number of mirrors that have been created. The number of mirrors can be dynamically modified as application needs change. The frequency of mirror updates can also be adjusted based on the needs of the application. The administrator can mirror just those segments of the global file system that require the additional bandwidth; there is no need to mirror the entire volume, as in some NAS systems. Mirrors can deliver nearly unbounded read bandwidth to any segment of the global namespace. This is highly suitable for many cluster-wide applications, including binary files commonly found in /usr/local or other common locations.

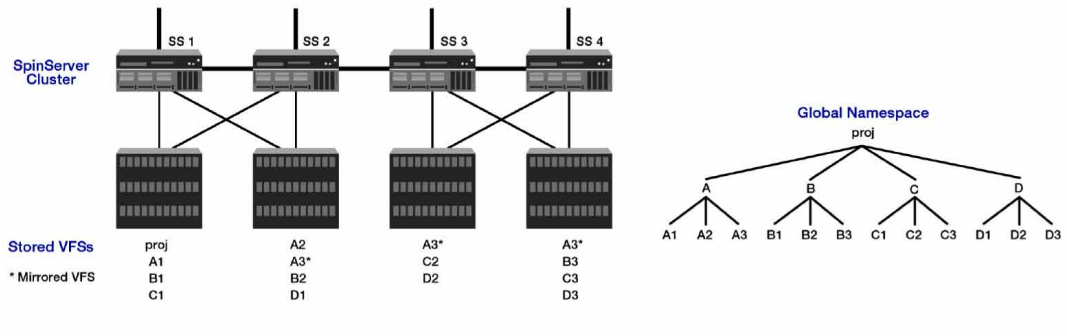

Figure 10) Mirroring A3 for enhanced performance.

7) SpinServer Validation by Independent Testing Lab

The Enterprise Storage Group, Inc. (ESG) recently performed a broad battery of evaluation tests on SpinServer. ESG Lab is a leading independent storage analyst firm and one of the most frequently quoted sources of storage industry intelligence. The evaluation tests performed by ESG Lab included:

- Scalability test
- Snapshot copies for file-level restore test
- Mirroring a virtual file system test
- Moving a virtual file system online test
- Windows, UNIX, and Linux sharing test
- NDMP backup support test
- SNMP and SMI-S support test
- Rolling code upgrade test
- Fault tolerance test
- ESG Lab performance and scalability audit

As a result of these tests, ESG Lab published an *ESG Lab Validation Report* on SpinServer, the results of which are summarized below. According to ESG, the company has:

> accomplished a Herculean task developing a scalable architecture, excellent performance. and a suite of features and functions required to compete against the major NAS incumbents. After evaluating SpinServer, analyzing performance data, and interviewing customers, ESG Lab considers this product to be an Enterprise NAS solution.

7.1 Major Findings

The major findings published by ESG Lab in its validation report include the following:

- SpinServer supports potentially limitless scalability, including storage capacity, CPU, memory, bandwidth, and performance
- SpinServer enables scalability by allowing nodes to be added nondisruptively into the cluster
- SpinServer allows for the management of multiple servers as a single pool of storage
- Adding nodes while online is an easy and transparent process that does not impact users
- SpinServer allows for nondisruptive system reconfiguration and file redistribution
- SpinServer provides flexible and cost-effective high availability via a server failover architecture
- SpinServer as a single node is significantly faster than major competitors
- SpinServer performance scales as more nodes are added
- SpinServer installation was easy, taking less than 10 minutes to create a VFS and share files
- Shared file access using NFS and CIFS was demonstrated
- Creating Snapshot copies was simple and flexible, while restoring Snapshot copies was quick and intuitive
- SpinServer can support thousands of Snapshot copies online
- It was simple to create local mirrors

- All files were copied to a node in the same cluster located at a remote site using asynchronous mirroring
- SpinServer has no single point of hardware failure and supports fault tolerance
- SpinServer supports high-availability server failover with a flexible set of configuration options, including 1+1, N+1, and NxM
- Rolling software code upgrades were performed while online and worked flawlessly
- Management alerts using SNMP were tested using HP OpenView to monitor

7.2 Scalability Test

7.2.1 Overview

One of the key advantages of SpinServer is the ability to add nodes easily and transparently within the same file system image or namespace. Customers require this capability in order to increase capacity and scale performance. ESG Lab simulated a customer environment with the following characteristics:

1. A file system used by Linux clients consumed nearly all available capacity
2. A new group of users on a different subnet required access to the file system
3. Performance was a concern as new users and applications were given access to the file system
4. Applications running against the file system could not be taken down for scheduled maintenance

7.2.2 Procedure

ESG Lab's procedure for this validation test was as follows:

1. Compilation and write/read/compare jobs were started on a Linux PC attached to the SpinServer cluster
2. A new SpinServer node was removed from production packaging and powered up
3. The new SpinServer node was cabled with Ethernet on the front end and FC for back-end connectivity
4. A third node was added to the cluster using the SpinServer GUI
5. Storage pools were configured using the newly added drives
6. While files were being accessed by the compilation and write/read/compare jobs, they were moved to the newly expanded storage space within the file system
7. The new location of the files was verified with the GUI
8. Background operations continued throughout the test without error
9. A new virtual interface was created on another subnet with access to the expanded Linux file system

7.2.3 Conclusions: Extremely Scalable

ESG Lab stated that traditional NAS systems have limited growth and scalability. When customers want to add more capacity to a single file system, traditional NAS systems have relatively limited resources. Customers are forced to create another file system and must support another device. Storage capacity is not the only limitation. An application may require more processing power and bandwidth, and the traditional NAS system may have insufficient resources to meet these needs.

In contrast, ESG Lab found that SpinServer is extremely scalable and can keep adding more storage capacity, processing power, and bandwidth to a single file system. This process is simple and quick: It takes less than an hour to add a new node. This simplifies management and can potentially save customers significant resources, time, and money.

ESG Lab identified the following key advantages of SpinServer scalability:

- Adding processing and storage capacity to a SpinServer cluster is easy and nondisruptive
- The SpinServer architecture provides a single management interface for a scalable clustered file server with a theoretical clustered limit of 512 nodes and 11PB of storage
- Traditional NAS servers require the creation of additional file systems and management points due to architectural limitations
- The SpinServer architecture is extremely scalable, with each additional node providing more CPUs and RAM for processing and caching; each additional drive array yields additional raw capacity and adds more drive actuators, which increases performance

7.3 Snapshot Copies for File-Level Restore Test

7.3.1 Overview

Snapshot copies are used for a variety of reasons, including quick recovery of deleted or corrupted data. ESG Lab created read-only Snapshot copies of a file system within a SpinServer cluster. This was followed by a user-level restore of a file from one of the Snapshot copies. ESG Lab simulated a customer environment with the following characteristics:

1. A storage administrator set up nightly Snapshot copies of a shared Windows NT® drive
2. Some time later a user created a Word document and sent it to a colleague for review
3. During the review cycle an important paragraph was accidentally deleted
4. The user replaced the original file with the reviewed version
5. The user noticed the missing paragraph, restored a Snapshot copy from last night, and retrieved the lost paragraph

7.3.2 Procedure

ESG Lab's procedure for this test was as follows:

1. A Snapshot interval of one minute was selected
2. A document was copied to a Windows drive letter mapped to the SpinServer cluster
3. A paragraph was added to the document
4. The Snapshot activity was viewed
5. The new paragraph was deleted
6. More Snapshot copies were observed
7. The Snapshot copies were accessed in a read-only file folder
8. Snapshot documents were opened to find the version containing the lost paragraph
9. The lost paragraph was recovered

7.3.3 Conclusions: Easy and Quick Restores

ESG Lab concluded that Snapshot copies are an essential and requisite function for an enterprise-class NAS system. They found that functionality worked as expected:

- Snapshot copies were easy to create
- Any or all files within the authorized directory structure of a protected VFS can easily and quickly be restored from disk without system administration intervention
- A large number of Snapshot copies per VFS provide flexibility well beyond the typical requirements of nightly incremental Snapshot copies over the course of a month

7.4 Mirroring a VFS Test

7.4.1 Overview

ESG Lab tested the SpinServer mirror function in order to mirror the contents of a VFS within a cluster. SpinServer creates a local copy of a VFS. Using this feature, customers can also create a remote mirror for business continuance purposes. Customers can add a node to the SpinServer cluster at a remote site and create a mirror of all the files in the cluster to that node.

Essentially, SpinServer allows customers to stretch the cluster and then create an asynchronous mirror to the remote node(s). This is both a local mirror (a copy within the same cluster) and a remote mirror (the node and disks in which the data is mirrored are at a remote site).

ESG Lab simulated a customer environment to test the mirroring of a VFS:

1. A SpinServer cluster was deployed between buildings within a campus
2. Engineering and customer service departments were located in different buildings
3. Engineering access is through the NODE01
4. The engineering file system containing source code was mirrored to NODE03 in customer service
5. A disaster occurred that temporarily shut down the engineering building (e.g., undetected water main break over a long weekend)
6. Engineers accessed the mirrored file system through NODE03 in customer service
7. Some time later the facility reopened
8. Mirroring was restored, and engineering NODE01 regained primary file system status

7.4.2 Procedure

ESG Lab used the following procedure for this test:

1. Background compilation and write/read/compare jobs were started from a Linux PC
2. The /gcc VFS containing source code on NODE01 was mirrored to NODE03
3. Mirror synchronization progress was observed using the SpinServer GUI
4. Once the mirror synchronization was complete (a couple of minutes), the mirror relationship was broken using SpinServer (simulating the disaster)
5. The mirrored file system was mounted and accessed from the Linux PC
6. The contents of the mirrored file system were compared to the primary file system
7. Compilation jobs were restarted using the mirror

7.4.3 Conclusions: Easy and Effective Mirroring

ESG Lab found that online mirroring with SpinServer is easy to perform. The unique SpinServer approach of stretching the cluster to remote sites is practical and works well. The product performs asynchronous mirroring, allowing for long-distance replication. Conclusions included:

- Online mirroring is easy to perform
- Mirroring supports both local and stretched cluster configurations
- Mirroring can be used for point-in-time imaging as well as for backup staging

7.5 Moving a VFS Online Test

7.5.1 Overview

ESG Lab tested the SpinServer VFS Move functionality, which allows an existing file system to be migrated while online and serving data within a SpinServer cluster. The VFS containing a Windows-based video was moved from one node to another.

7.5.2 Procedure

ESG Lab's procedure for this test was as follows:

1. Two background jobs were started on the SpinServer cluster: a video application on an attached Windows PC and a GNU C compilation on an attached Linux PC
2. A VFS Move operation was performed using the SpinServer GUI
3. The VFS containing the Windows video was moved from NODE01 to NODE03
4. The VFS containing the GNU C source code was moved from NODE03 to NODE01
5. ESG Lab observed that both applications kept running
6. The new configuration and migration activity were observed using the GUI

7.5.3 Conclusions: Transparent Moves

ESG Lab concluded that customers can move data within a SpinServer cluster easily and transparently. The Move functionality is easy to use and allows for online data movement within a SpinServer cluster. Customers can move file systems to faster drives for better performance or to slower and less expensive drives as part of an information lifecycle management (ILM) strategy. Conclusion included:

- Moving virtual file systems was easy to perform
- Movement was nondisruptive and an online process for both NFS and CIFS clients
- No changes to mount points or directory structures were required

7.6 Windows, UNIX, and Linux Sharing Test

7.6.1 Overview

ESG Lab stated that NAS solutions should support Windows, UNIX, and Linux operating systems, allowing clients from these environments to share the same data. ESG Lab used the SpinServer GUI to create VFSs on the SpinServer cluster. The VFSs were spread physically across the cluster, but all were easily navigated from a single Windows drive letter mapped to the SpinServer cluster. The VFSs were easily configured for sharing across Windows, UNIX, and Linux clients.

7.6.2 Procedure

The procedure used by ESG Lab for this test was as follows:

1. ESG Lab created a VFS called gcc
2. ESG Lab then accessed data on the VFS from Windows NT and Linux clients
3. The Windows NT client used CIFS and accessed data in the cluster as drive letter F:
4. The Linux client accessed data using the NFS protocol as a mounted directory (/gcc)

7.6.3 Conclusions: Intuitive Interoperability

- Creating a VFS on the SpinServer cluster was easy and intuitive
- SpinServer provides a unified solution for shared file access by Windows, UNIX, and Linux clients

7.7 NDMP Backup Support Test

7.7.1 Overview

The Network Data Management Protocol (NDMP) is an open standard protocol for enterprise-wide backup of heterogeneous NAS storage. Using NDMP, a backup application can communicate with the SpinServer cluster over an Ethernet interface to direct backup traffic to a tape device attached to the SAN or a network-attached backup server. ESG Lab performed an NDMP backup of Linux and Windows file systems using VERITAS NetBackup.

7.7.2 Procedure

ESG Lab's procedure for this validation test was as follows:

1. VERITAS NetBackup V4.5 was installed on a Windows 2000 PC
2. NDMP was enabled, and a user ID and password were created using the SpinServer GUI
3. A CIFS dynamic disk was accessed as the F: drive, and a Linux file system was backed up

7.7.3 Conclusions: Support of NDMP Backups

ESG Lab concluded that SpinServer fully supports NDMP, and customers can expect to run their backups as they would with other NAS products.

7.8 SNMP and SMI-S Support Test

7.8.1 Overview

ESG Lab evaluated SpinServer SNMP and error notification support.

7.8.2 Procedure

The following procedure was used to test SpinServer SNMP support:

1. HP OpenView V6.1 was installed on a Windows 2000 PC
2. The SpinServer SNMP agent was installed on the Windows 2000 PC
3. SNMP was enabled on the SpinServer cluster, and the IP address was configured using the SpinServer GUI
4. E-mail notification was set up to send e-mails to ESG Lab for critical errors
5. E-mail and HP OpenView notification (red boxes) were noted when faults were injected
6. The SpinServer GUI was launched successfully from HP OpenView to drill down into configuration information and to diagnose errors

The Storage Networking Industry Association (SNIA) launched the Storage Management Initiative (SMI) in 2002 with a goal of creating an SMI Specification (SMI-S), which defines an open interface for the management of storage networks.

7.8.3 Conclusions: Support of Open Standards

- The SpinServer command line interface and GUI are designed to be SMI-S compliant
- SpinServer supports open management standards
- SpinServer avoids proprietary interfaces
- SpinServer is designed to enable integration with other management solutions
- SpinServer integrates well with other vendors' equipment

7.9 Rolling Code Upgrade Test

7.9.1 Overview

ESG Lab found that SpinServer supports online rolling code upgrades. This process is performed in the SpinServer cluster by running a controlled failover of each node, upgrading the code on that node while it is not part of the active cluster. The offline node is then brought back online, and the process continues to the next node.

7.9.2 Procedure

ESG Lab did not perform a rolling code upgrade, but rather interviewed customers that had.

7.9.3 Conclusions: Painless Upgrading

- Upgrading code revisions should be a painless process that causes no downtime to the users
- The SpinServer architecture is well suited for online rolling code upgrades
- Customers interviewed by ESG Lab have reported that this capability is easy and trouble-free with SpinServer

7.10 Fault Tolerance Test

7.10.1 Overview

A series of faults were injected during validation testing to ensure that a SpinServer cluster configured for high availability could survive a series of errors. A write/read compare script and a Windows PC running a video were started from Linux and NT PCs attached to the SpinServer cluster. ESG Lab created a high-availability SpinServer cluster using the three-node cluster used in previous testing. SpinHA® support of 1+1 and N+1 server failover configurations was validated during this testing.

7.10.2 Procedure

The following errors were injected as background jobs ran without incident:

1. Pulled a redundant Ethernet connection from the SpinServer cluster to the Gigabit Ethernet switch
2. Pulled redundant FC cables from the SpinServer cluster–to-FC switch and to the drive enclosure
3. Pulled a drive in a RAID set and observed a rebuild beginning
4. Pulled a mirrored system drive from a SpinServer cluster
5. Powered off the SpinServer controller and drive enclosure

This stage of testing was concluded by reattaching all cables and powering up all components. SpinServer was used to "fail back" to a fully clustered and protected configuration. Once the system was recabled and powered up, a manual process was used to return the SpinServer cluster to a fully protected state. The requirement for user intervention after restoration in a clustered configuration is a correct and industry-accepted practice.

7.10.3 Conclusions: High Availability

ESG Lab stated that a requisite requirement for a true enterprise-class NAS solution is supporting high availability. The lab concluded that SpinServer can be configured as a highly available cluster, ensuring no single point of hardware failure.

7.11 ESG Lab Performance and Scalability Audit

ESG Lab audited SpinServer performance benchmark results and methodologies to validate claims that the SpinServer architecture delivers industry-leading performance that scales beyond that available from other NAS products on the market today.

The lab consulted the SPEC Web site (*www.spec.org*) for published SPEC SFS results. They determined that a fair comparison of SpinServer to market-leading midrange controllers yields a 40% performance advantage. A five-node SpinServer cluster achieved an outstanding rate of over 130,000 networked file system I/O operations, which is more than twice as fast as any published benchmark result from any vendor to date.

After a thorough investigation of published SPEC SFS results and a review of internal engineering performance results, ESG Lab stated that they support the claims that SpinServer is the highest performance NAS product compared to the other products tested by SPEC SFS.

In addition to impressive SPEC SFS results, ESG Lab concluded that SpinServer should perform well in single stream and aggregate stream tests performed on very large files. ESG Lab reviewed results from internal company tests with large files, and the numbers were impressive. Large file benchmarks are important measures of performance for applications in the oil and gas, entertainment, life science, and engineering industries.

7.12 Customer Feedback

ESG Lab interviewed SpinServer customers to get their insight and feedback on SpinServer. Some of the quotes reported by the lab were:

- "We have a dynamic environment. We are creating lots of large and small files; we are doing lots of creations and deletions. Our SpinServer cluster has millions of files stored on it, and the performance has been excellent."
- "We have a Linux clustered environment with 64 processors running on 32 machines. There are approximately 100 regular users. We are a high-performance environment, and the SpinServer cluster handles anything we throw at it."

7.13 ESG Lab Conclusions

ESG Lab reported the following bottom-line conclusions:

- SpinServer is an enterprise-class NAS product that combines sophisticated software functionality to provide an extremely scalable, high-performance, and feature-rich solution
- SpinServer is well suited for both large and midrange environments
- SpinServer supports all requisite NAS functionality, such as NFS/CIFS support, Snapshot copies and restores, mirroring, remote mirroring, and high availability

8) Conclusion

Linux-based compute grids are emerging as the primary choice for enterprise-level, high-performance computing applications. Across many industries and applications, Linux compute farms provide a cost-effective, modular solution that is able to scale computing performance to virtually any level.

Today's grid computing enterprise requires a back-end storage solution that can keep pace with growing and changing demands for data. The solution must scale performance of data to levels well beyond that which a single storage system can deliver. What is needed is a flexible grid storage solution that can manage large data sets reaching hundreds of terabytes—and can distribute the data across countless servers. Faced with data sets that are constantly growing, the grid computing enterprise needs a storage solution with the ability to expand capacity and rebalance data without downtime.

Independent lab tests have validated that NetApp SpinServer is a flexible, high-performance storage solution that meets these critical challenges. The SpinServer solution eliminates storage hot spots by transparently moving data from an overburdened server to an underutilized one. It provides a global namespace for all data stored in the cluster. SpinServer is an extremely scalable solution that makes it easy to add more storage capacity and bandwidth. The net result is a simplified data management environment that can save the enterprise significant resources, time, and money.