

Massive High-Performance Global File Systems for Grid computing

Phil Andrews, Patricia Kovatch, Christopher Jordan
San Diego Supercomputer Center, La Jolla CA 92093-0505, USA
{ andrews, pkovatch, ctjordan }@sdsc.edu¹

Abstract

In this paper we describe the evolution of Global File Systems from the concept of a few years ago, to a first demonstration using hardware Fibre Channel frame encoding into IP packets, to a native GFS, to a full prototype demonstration, and finally to a production implementation. The surprisingly excellent performance of the Global File Systems over standard TCP/IP Wide Area Networks has made them a viable candidate for the support of Grid Supercomputing. The implementation designs and performance results are documented within this paper. We also motivate and describe the authentication extensions we made to the IBM GPFS file system, in collaboration with IBM.

In several ways Global File Systems are superior to the original approach of wholesale file movement between grid sites and we speculate as to future modes of operation.

1. Introduction

At the beginning of this century, Grid Computing¹ was beginning to make inroads on production systems. The general approach was to take the Globus software² developed originally within academic environments and move it wholesale to the target systems. The TeraGrid³ was probably the first really extreme example, with a network backbone capable of 40 Gb/s, compute capability of well over 10 Teraflops, rotating storage of over a Petabyte, and archival capacity in the 10's of Petabytes spread over several sites. It is a testament to the robustness of the original design that much of the translation from relatively constrained academic environments to very large scale production systems was successful. There is one specific area, however, where the original paradigm invited modification for efficient computations in the supercomputing arena. The original mode of operation for Grid Computing was to submit the user's job to the ubiquitous grid, where it would run on the most appropriate computational platform available. Any data required for the computation would be moved to the chosen compute facility's local disk, and output data would be written to the same disk; being saved to the user's permanent storage facility at the end of the job. The normal utility used for the data transfer would be GridFTP⁴.

¹ Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SC|05 November 12-18, 2005, Seattle, Washington, USA
(c) 2005 ACM 1-59593-061-2/05/0011...\$5.00

There were several reasons why the normal approach of moving data back and forth did not translate well to a supercomputing grid, mostly relating to the very large size of the data sets used. For example, the National Virtual Observatory (NVO)⁵ consists of approximately 50 Terabytes and is used as input by several applications. Some applications write very large amounts of data, e.g., the Southern California Earthquake Center (SCEC)⁶ simulations may write close to 250 Terabytes in a single run. Other applications require extremely high I/O rates in order to read and write intermediate results: the Enzo⁷ application requires multiple Terabytes per hour be routinely written and read.

These sizes and required transfer rates are not conducive to routine migration of wholesale input and output data between grid sites. The computational system chosen may not be able to guarantee enough room to receive a required dataset, or for the output data, while the necessary transfer rates may not be achievable. Resource discovery⁸ can go a long way to help, but it may also prevent several sites from participating in the computational effort. In addition, in many cases the application may treat the very large dataset more as a database, not requiring anywhere near the full amount of data, but instead retrieving individual pieces of very large files.

In the case of Enzo, for example, many sites work to interpret and visualize the output data, and moving very large files to multiple sites may be both inefficient and restrictive on the participants.

In this paper we shall show how a Global File System (GFS), where direct file I/O operations can be performed across a Wide Area Network can obviate many of these

objections to grid supercomputing. One of the reasons that Global File Systems were not considered as part of the original TeraGrid is that while the middleware components of Globus can be developed and implemented largely by independent software developers, Global File Systems require close integration with both systems designers and vendor software developers. This necessitates a very measured implementation strategy, as missteps could easily have disastrous consequences for large numbers of users. In this vein, it was decided to thoroughly develop the concept via a series of large-scale demonstrations before attempting to integrate it into the TeraGrid infrastructure.

2. GFS via Hardware Assist: SC'02

In 2002, Global File Systems were still only in the concept stage, and there were real concerns that the latencies involved in a widespread network such as the TeraGrid would render them inoperable. Plus, the file systems themselves did not yet have the capability of exportation across a WAN. It was, however, possible to “fool” the disk environment by using recently developed specialist hardware capable of encoding Fibre Channel frames within IP packets (FCIP). In that year, the annual Supercomputing conference⁹ was Baltimore and with the San Diego to show floor distance being greater than any within the TeraGrid, this seemed the perfect opportunity to demonstrate whether latency effects would eliminate any chance of a successful GFS at that distance.

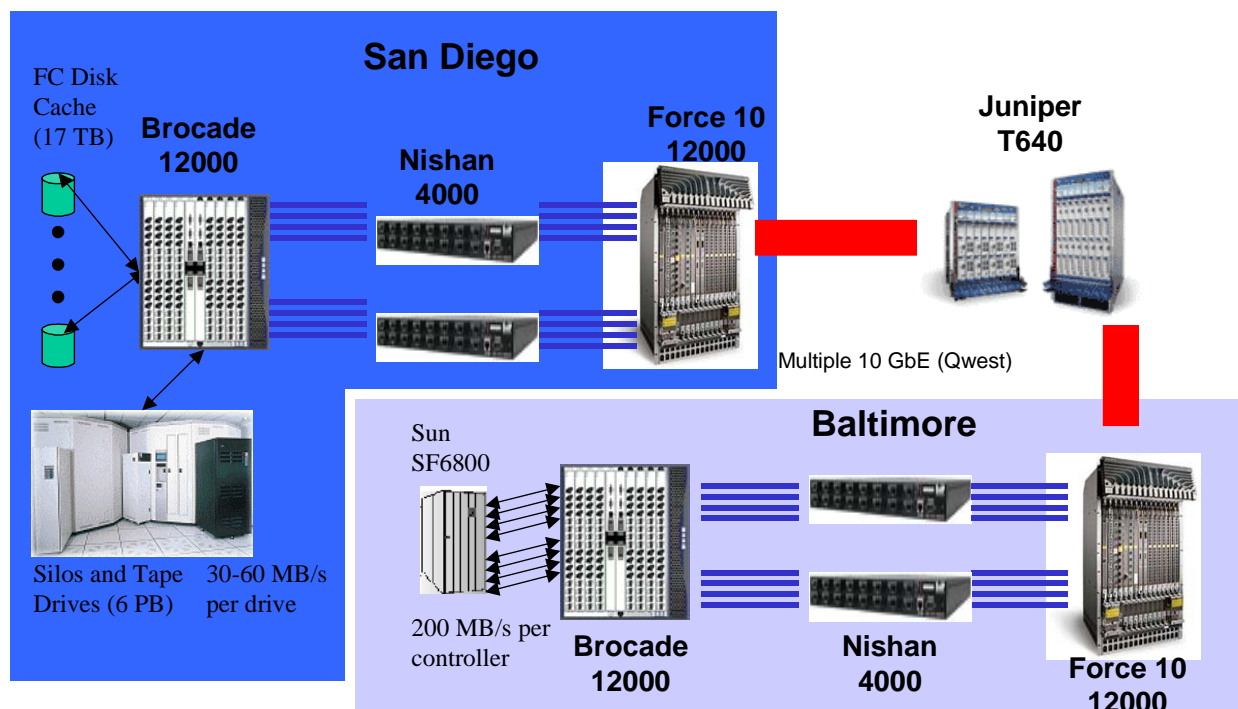


Figure 1. SC'02 GFS hardware configuration between San Diego and Baltimore

A standalone mass storage system was instituted at the San Diego Supercomputer Center (SDSC)¹⁰ with about 30 Terabytes of disk controlled by a Sun F15K server using Sun's QFS file system integrated with the SAM Hierarchical Storage Management software. The SANergy¹¹ software suite was used to export the QFS file system across a Storage Area Network. Locally, at SDSC, the SAN was managed by a Brocade 12000 Fibre Channel Switch. Using the TeraGrid backbone, plus a SciNet¹² managed extension from Chicago to Baltimore, a 10 Gb/s Wide Area Network was established from SDSC to the SC'02 show floor in Baltimore.

The actual exportation of the file system across the WAN was accomplished using two pairs of Nishan¹³ 4000 boxes which effectively encoded Fibre Channel frames into IP packets for transmission, and decoded them during reception. This extended the Storage Area Network across the country to the SDSC booth on the SC'02 show floor where another Brocade 12000 and a Sun SF6800 formed the Eastern part of the WAN-SAN configuration.

The connection between the Nishan 4000 boxes and the Force10 GbE switches was via 4 GbE channels, so the maximum possible bandwidth was 8 Gb/s.

With this being the first demonstration at this scale, performance could not be easily predicted, but in the event the transfer rate achieved was over 720 MB/s; a very healthy fraction of the maximum possible. It also demonstrated for the first time a property of Global File Systems that has become more familiar since: the very sustainable character of the peak transfer rate.

It not only demonstrated that the latencies (measured at 80ms round trip SDSC-Baltimore) did not prevent the Global File System from performing, but that a GFS could provide some of the most efficient data transfers possible over TCP/IP.

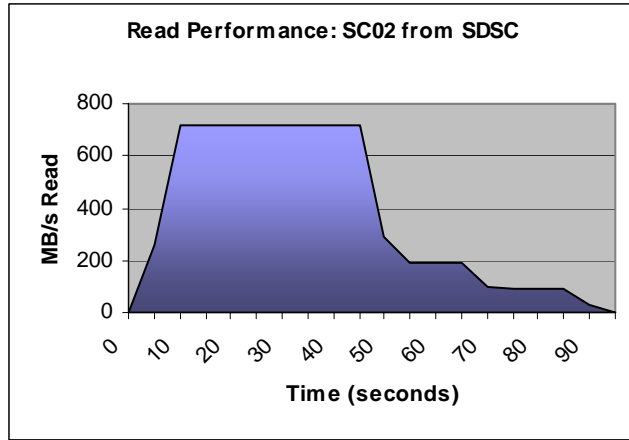


Figure 2. SC'02 GFS Performance between SDSC and Baltimore

3. Native WAN-GFS: SC'03

Having demonstrated that continent-wide latencies were not an insurmountable obstacle to Global File Systems, there was considerable interest in whether they

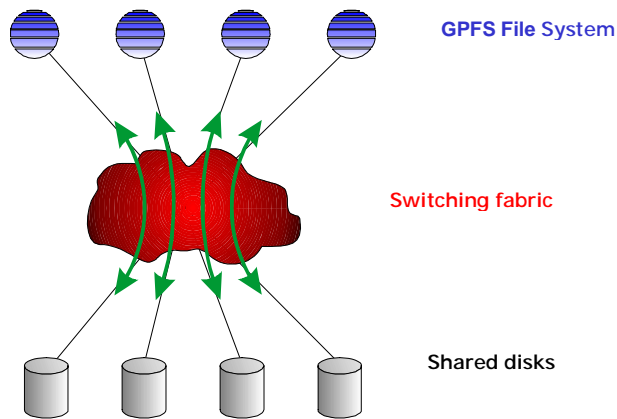


Figure 3. GPFS Organization

were possible without hardware FCIP encoding. The location of Phoenix, Arizona, for the show location did not provide another latency-challenging opportunity, but

the chance to use pre-release software from IBM's General Parallel File System (GPFS)¹⁴ that was a true wide area-enabled file system was irresistible. There are other parallel file systems, and a more detailed evaluation than is possible in this paper has already taken place¹⁵.

GPFS was a strong candidate for a WAN GFS because of its organization (shown in Fig. 3) where the disks were expected to be separated from the file system clients by a switching fabric of some kind. Normally, the fabric would be the interconnect of the machine itself, but TCP/IP was one of the supported protocols so that extension to a WAN interconnect was, conceptually at least, simple.

This demonstration would have been impossible without close collaboration with the GPFS development team at IBM, and their provision of pre-release software. In this case, three sites were involved: SDSC, the SDSC booth at the SC'03 show floor in Phoenix, and the National Center for Supercomputing Applications (NCSA)¹⁶ in Illinois. A 10GbE connection was provided by SciNet from the show floor to the TeraGrid backbone. This time, the central GFS was actually in the SDSC booth, where 40 two-processor IA64 nodes were used to serve the file system across the wide area network to SDSC and NCSA. The organization of the demonstration is shown in Fig. 4.

The mode of operation was to copy data produced at SDSC across the WAN to the disk systems on the show floor, then to visualize it at both SDSC and NCSA. The 40 IA64 servers each had a single FC HBA and GbE connectors, so there was sufficient bandwidth available to saturate the 10 GbE link from the show floor.

At SDSC the visualization was performed on another 32 IA64 servers, so this was a reasonable test of the ability of GPFS to exercise the WAN capability, at least at up to the 10 GbE level.

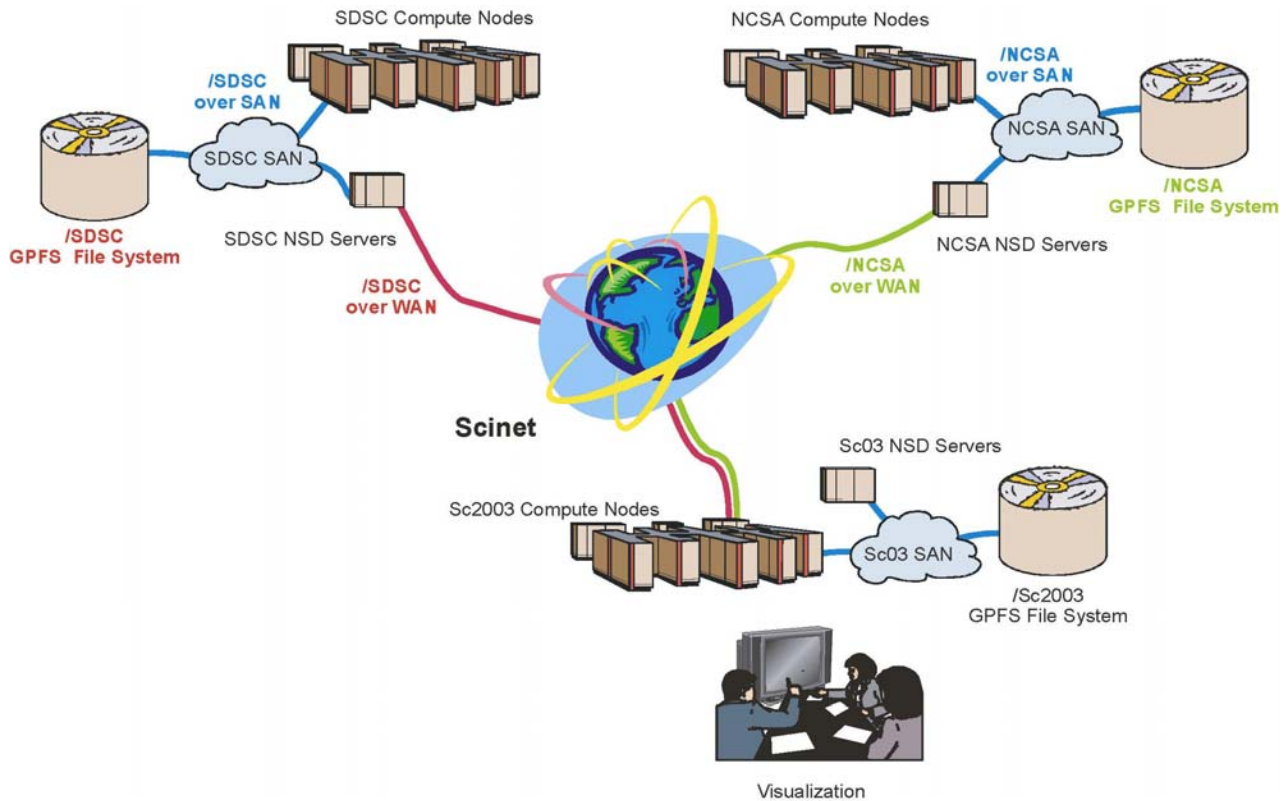


Figure 4. WAN-GPFS Demonstration at SC'03

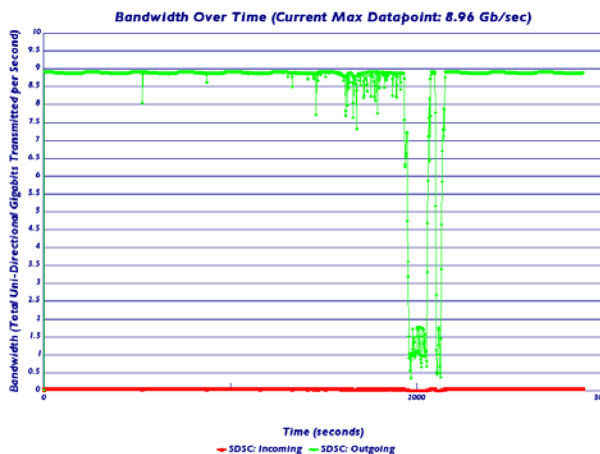


Figure 5. Bandwidth results at SC'03

The resulting bandwidth attained was very encouraging; over a maximum 10 Gb/s link, the peak transfer rate was almost 9 Gb/s (actually 8.96 Gb/s) and over 1 GB/s was easily sustained. The dip in Fig. 5 corresponds to the visualization application terminating normally as it ran out of data and was restarted.

4. True Grid Prototype: SC'04

Our experience at SC'02 and SC'03 gave us confidence that a very large, very high performance Global File System was a viable way to facilitate grid supercomputing. But such an approach is very different from just introducing grid middleware into an existing infrastructure; it requires changes in the infrastructure itself, significant financial investment, and the commitment of multiple sites. To achieve sufficient community confidence to proceed in this direction within the TeraGrid, it was decided that SC'04 would be the opportunity to implement a true grid prototype of what a GFS node on the TeraGrid would look like. Fortunately, SC'04 was the advent of the StorCloud challenge¹⁷ and significant amounts of vendor equipment were available for the demonstration.

It should first be stated what we were trying to emulate: the TeraGrid organization as it existed in early 2004 is shown in Fig. 5. The backbone between Los Angeles and Chicago is 40 Gb/s with each of the sites connected at 30 Gb/s.

SciNet was able to provide a 30 Gb/s connection from the show floor to the TeraGrid backbone, emulating the speed at which individual sites were connected to the TeraGrid. As part of the StorCloud initiative, IBM pro-

vided approximately 160 TB of disk located in the Stor-

Cloud booth on the Pittsburgh SC show floor.

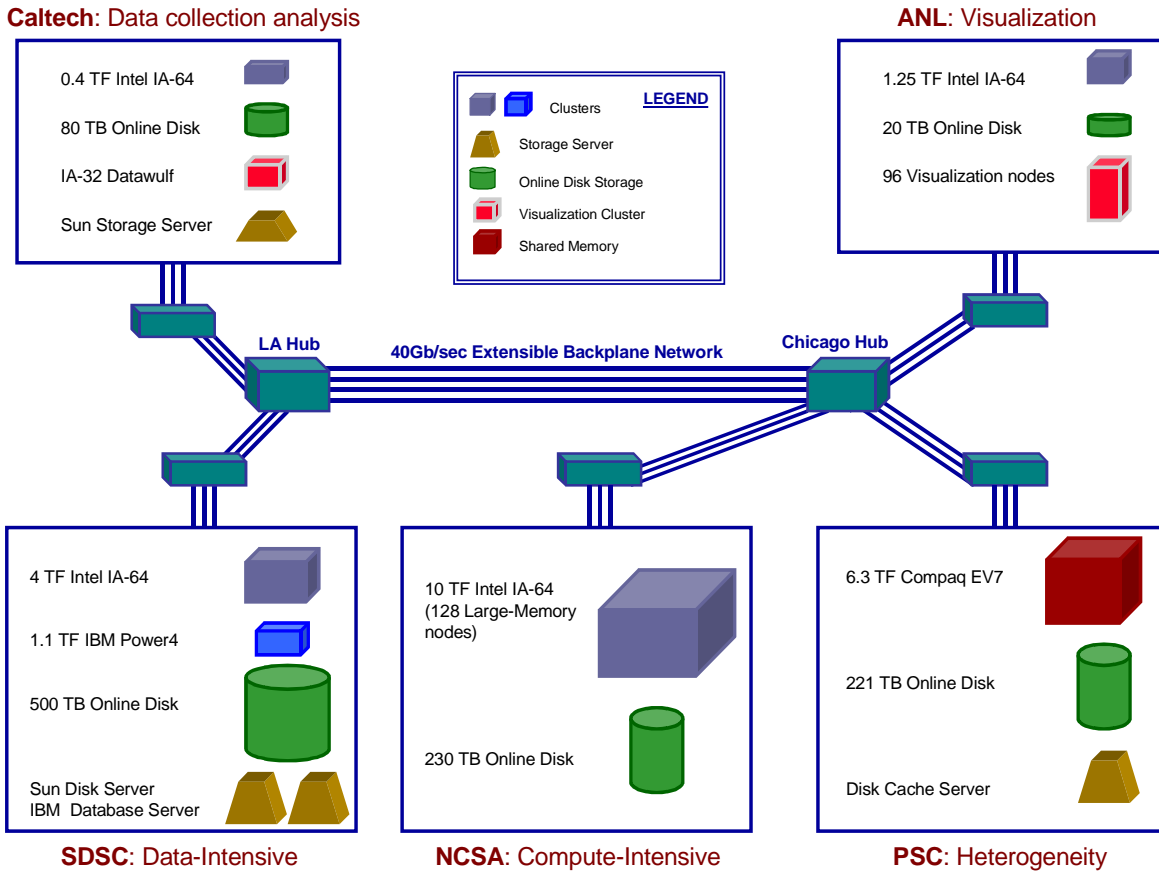


Figure 6. TeraGrid Organization as of early 2004

Once again, SDSC shipped 40 two-way IA64 nodes to the SDSC booth at SC in Pittsburgh, but this time each node had 3 Fibre Channel Host Bus Adapters and 120 two Gb/s FC links were laid between the SDSC and StorCloud booths. Total theoretical aggregate bandwidth between the disks and the servers was 240 Gb/s, or approximately 30 GB/s. In actual fact, approximately 15 GB/s was obtained in file system transfer rates on the show floor. Each of the 40 server nodes had a GbE connection to the SciNet network and the GPFS (again, a pre-release from IBM) was served up from the show floor to SDSC and NCSA. The attempt here was not so much to demonstrate the highest possible numbers, but rather to emulate true grid supercomputing as we expect it to be used in a data-intensive environment. After the arrangement was implemented, but before the show proper began, the application Enzo ran on the DataStar computational system at SDSC, writing its output directly the GPFS disks in Pittsburgh. This was an attempt to model as closely as possible what we expect to be one of the dominant modes of operation for grid supercomput-

ing: the output of a very large dataset to a central GFS repository, followed by its examination and visualization at several sites, some of which may not have the resources to ingest the dataset whole.

The I/O requirements of the Enzo application as it wrote the output data, while significant, are on the order of a Terabyte per hour and did not stress the 30 Gb/s capability of the TeraGrid connection. The post-processing visualization, however, was chosen to be largely network limited in order to discover just how quickly the GFS could provide data in a scenario that mimicked as closely as possible the expected behavior of a production system on the TeraGrid. For a complete examination, we also used a simple sorting application that merely sorted the data output by Enzo, and was completely network limited. This was run in both directions, to look for any differences in reading and writing while connected to the central GPFS GFS on the Pittsburgh show floor.

The overall organization of the demonstration is shown in Figure 7.

SC '04 Demo IBM-SDSC-NCSA

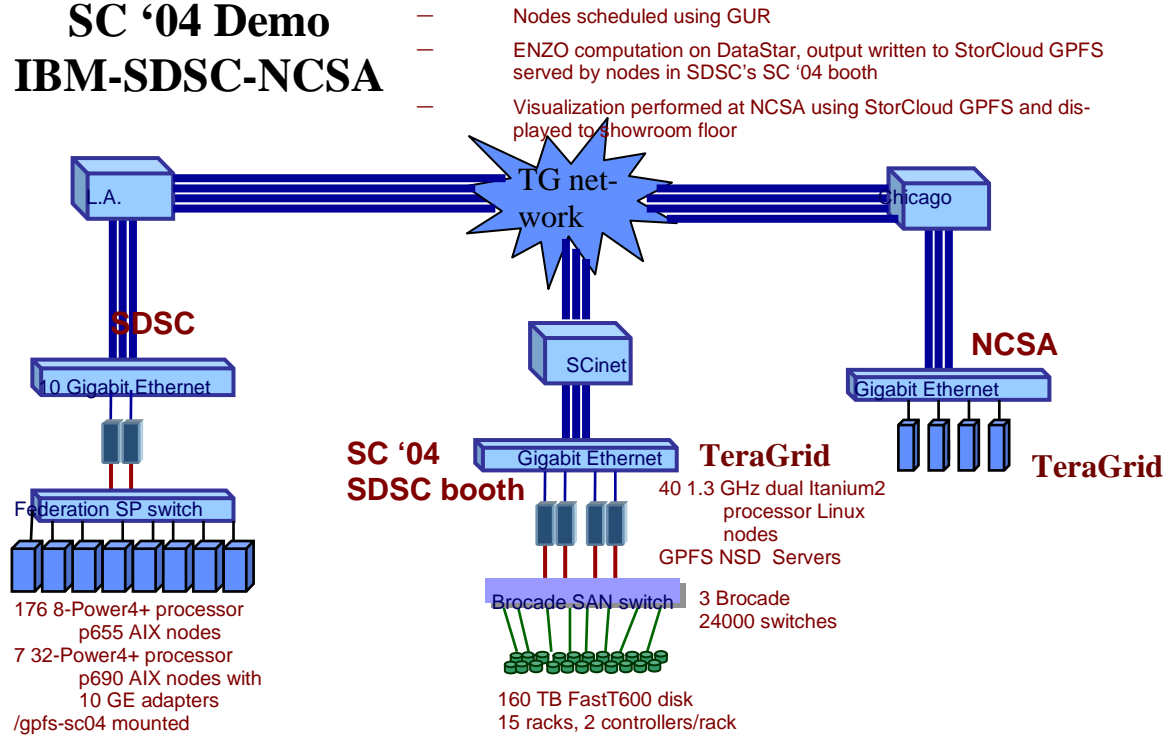


Figure 7. Prototype Grid Supercomputing at SC'04

Performance during network limited operations was very encouraging. As part of the SciNet Bandwidth challenge at SC'04, the SciNet team monitored and documented the bandwidth used by the demonstration during its high transfer rate stage.

Each of the three 10 Gb/s connections between the show floor and the TeraGrid backbone were monitored separately and are displayed in Figure 8., as is the aggregate performance. Individual transfer rates on each 10 Gb/s connection varied between 7 and 9 Gb/s, with the aggregate performance relatively stable at approximately

24 Gb/s (3 Gb/s). The momentary peak was over 27 Gb/s, sufficient to win this portion of the Bandwidth challenge. Both reads and writes were demonstrated, in an alternate manner, but the rates were remarkably constant. Rates between the show floor and both NCSA and SDSC were virtually identical. It should also be noted that this was the first time that an SDSC extension to the GPFS WAN approach was used: true GSI authentication which we will describe later in this paper.

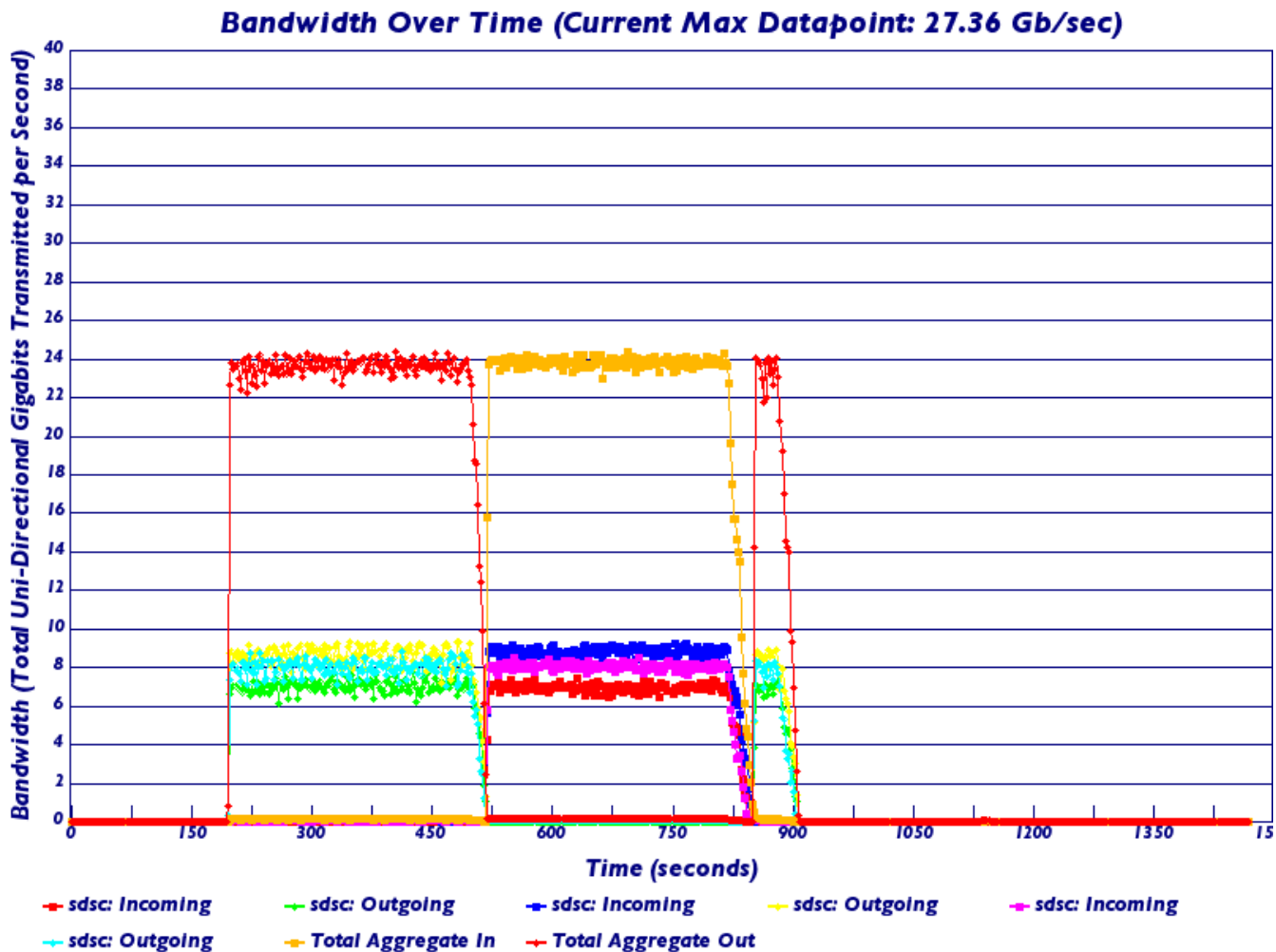


Figure 8. Transfer Rates at SC'04

5. Production Facility: 2005

The experiences of SC'02, SC'03, and particularly SC'04 were positive enough to convince SDSC, TeraGrid, and NSF personnel to proceed towards production infrastructure using this approach. By this time, the size of datasets had become large enough that only massive installations really held promise for significant facilitation of grid supercomputing. In fact, the size of some datasets was already starting to stress the storage capabilities of several sites. The NVO dataset, for example, was proving particularly useful and multiple sites were committed to providing it to researchers on spinning disk.

At 50 Terabytes per location, this was a noticeable strain on storage resources and if a single, central, site could maintain the dataset this would be extremely helpful to all the sites who could access it in an efficient

manner. Of course, updates, data integrity, backups, etc., could also be handled in a much more satisfactory way if copies were not spread around the TeraGrid.

Given that NVO at 50 Terabytes was only one such dataset, it was clear that for significant impact, a very large amount of spinning disk (by early 2005 standards) would be required. The onset of Serial ATA (SATA) disk drives made this more of a financial possibility, and in late March, 2005, approximately 0.5 Petabytes of SATA disk was acquired by SDSC. If this installation is successful, the intention is acquire another 0.5 Petabytes of disk by October 2005, with up to 1 Petabyte available for a Global File System.

Experience shows that is essential to design a balanced configuration, particularly with this much disk. It was decided that a design point would be for an eventual maximum theoretical bandwidth within the SDSC machine room of 128 Gb/s. This number should be easily sufficient to fill the SDSC path to the TeraGrid backbone, and is an exact match to the maximum I/O rate of our

IBM Blue Gene/L system, “Intimidata”, which is also planned to use the GFS as its native file system, both for convenience and as an early test of the file system capability. For the Network Shared Disk (NSD) servers we used 64 two-way IBM IA64 systems with a single GbE interface and Fibre Channel 2 Gb/s Host Bus Adapter in each. The plan is to double both of these in the future: the GbE to increase the maximum aggregate throughput to 128 Gb/s and the HBA to provide a separate path for an archive interface to the disk storage.

The disks themselves are 32 IBM FastT100 DS4100 RAID systems, with 67 250 GB drives in each. The total raw storage is thus $32 \times 67 \times 250 \text{ GB} = 536 \text{ TB}$. The RAID sets within the DS4100 are internally connected via two 2 Gb/s Fibre Channel arbitrated loops, with an independent controller for each loop. The drives within each RAID set are Serial ATA disks. The two DS4100 controllers each have a 2 Gb/s FC connection to the outside world so that each DS4100 system can connect to two NSD servers. The disk setup for a single DS4100 is displayed in Fig. 9 with seven 8+P RAID sets shown. The remaining unused drives function as hot spares.

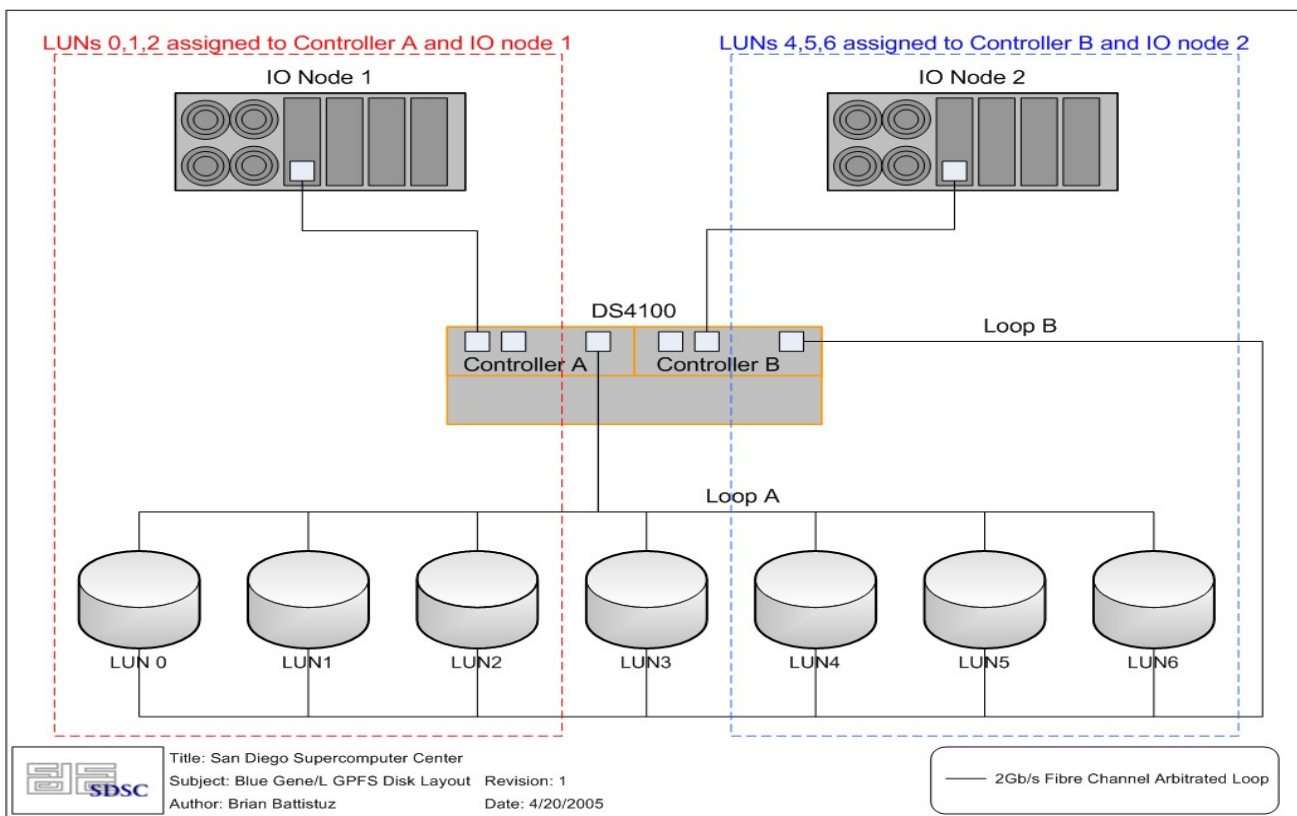


Figure 9. SATA disk arrangement

Although the connection between the DS4100 systems and the I/O servers is via Fibre Channel, the outside world normally accesses the file system data via the GbE connection to each I/O server. For maximum usability, the file system needs to be accessible from the systems in the SDSC machine room as well as across the TeraGrid Wide Area Network, and Fig. 10 shows how the 0.5 Petabyte of disk is accessed both (presently) by NCSA and Argonne National Laboratory, and by the major compute resources of SDSC. We have recently begun

semi-production use of the approximately 0.5 PB of GFS disk, mounting them in a production manner at some nodes at several sites: all 256 nodes of the TeraGrid Cluster at SDSC, all 32 nodes at Argonne National Laboratory, and a few nodes at Indiana University and NCSA. We have some preliminary performance numbers, at ANL the maximum rates are approximately 1.2 GB/s to all 32 nodes. We were able to perform a scaling investigation at SDSC, and we show the results in Fig. 11. Although the network communication is similar to data serving to a much more geographically remote center, it

must be remembered that this is still within the same machine room, with no TG backbone involvement. Even so, the performance measured is gratifying; with a measured maximum of almost 6 GB/s, within a network environment with a theoretical maximum of 8 GB/s. The observed discrepancy between read and write rates is not yet understood, but is not an immediate handicap since

we expect the dominant usage of the GFS to be remote reads. At the moment, no remote sites have enough nodes mounted to stress the capabilities of the file system, but we hope to change this by the intended production date of Oct 1, 2005.

SDSC Local and WAN GPFS Configuration

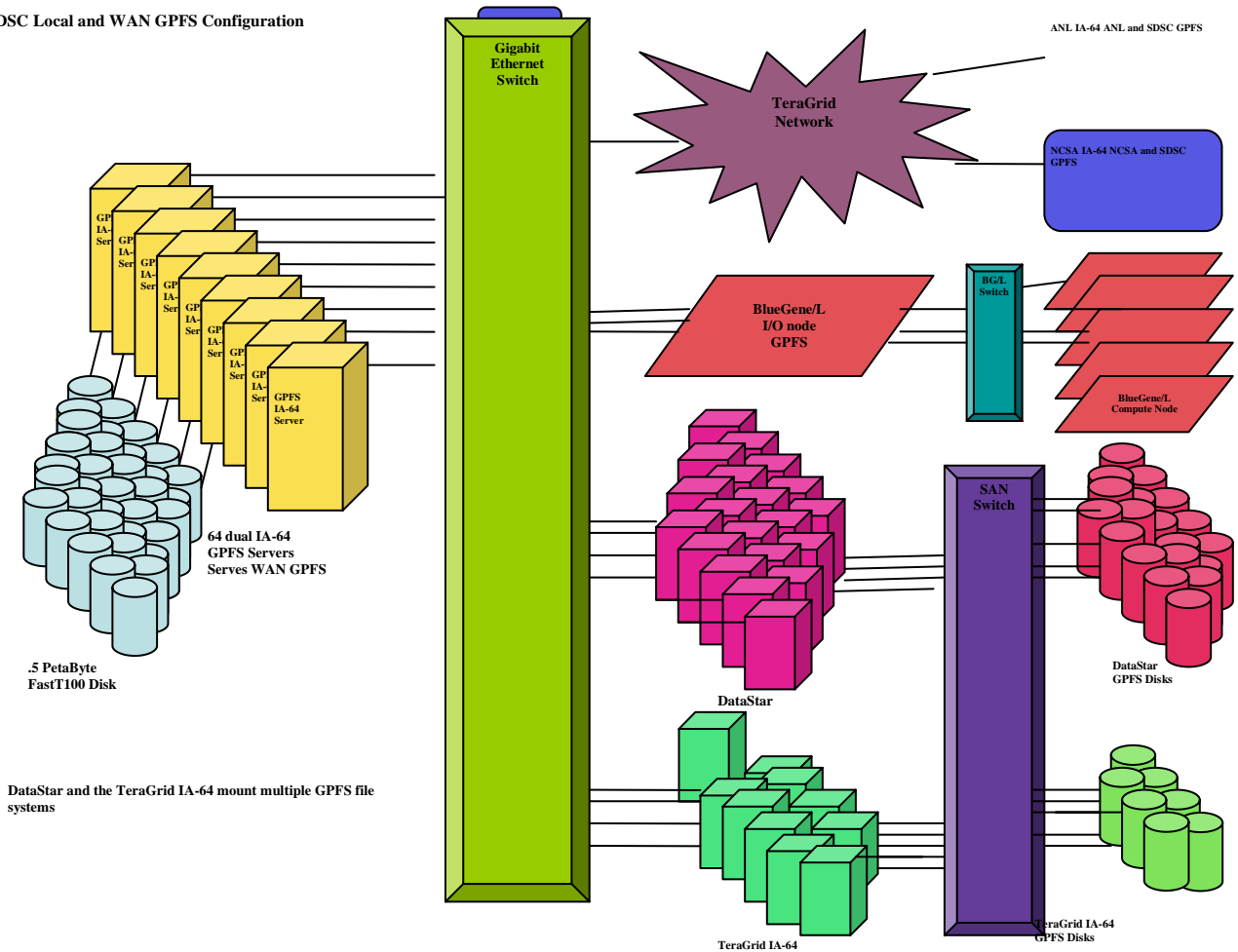


Figure 10. Network organization

MPI IO, 128 MB Block Size, 1 MB Transfer Size

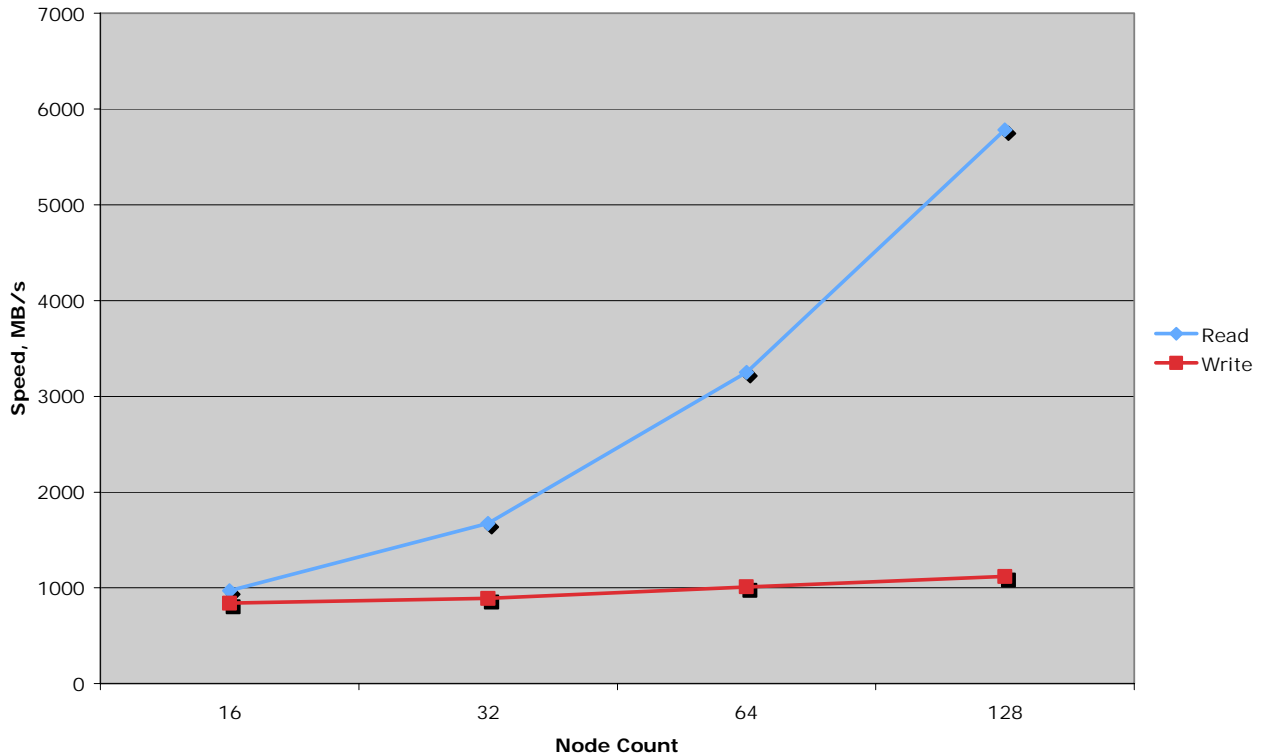


Figure 11. Performance scaling with number of remote nodes, Reads and Writes

6. Authentication

So far in this paper, we have concentrated on working with vendors in the development and early testing of external software, on the assumption that for true production quality performance and support, the development of new software should be minimized. There was one area, however, where the production of new capabilities was unavoidable for the adoption of this approach. That was in the field of authentication. The existing protocol for file system authentication is via the User ID (UID), and secondarily by Group ID (GID). These are normally uniform within an organization, so, e.g., every SDSC user has the same UID on each of the SDSC systems. This greatly simplifies access mechanisms, etc., and is the conventional approach for file systems. The difficulty enters when attempting to extend the concept of file “ownership” across a Grid. Except in very tightly coupled cases, it is very likely that a user will have different UID within the separate organizational entities making up a Grid. E.g., a user will, most likely, have different UIDs at SDSC, NCSA, ANL, etc., across the TeraGrid. However, he will certainly prefer to believe that any data he creates on a centralized Global File System belongs to “him” and not to one of his particular accounts, which

probably has no logical relation to his account on any other Grid system. In response to this dilemma, we decided to develop an extended authentication mechanism based on the GSI¹⁸ model, where a single certificate allows a user access to multiple computation platforms. Since we expected each of the TeraGrid users to have access to a GSI certificate, we decide to use this approach for data access, also. We describe our work in the following subsections.

6.1 GPFS 2.3 Cluster Control

A GPFS “cluster” is simply a set of nodes which share configuration and local filesystem information, and which can be controlled as a whole using the standard GPFS commands. Cluster configuration and filesystem consistency is maintained by a primary, and optionally a secondary, configuration server, which maintains master copies of all configuration files. In some cases, simple configuration queries can be serviced by filesystem daemon network requests to the primary configuration server; in these cases, internal authentication mechanisms are used based on the local configuration files on each node. However, when a collective command is issued, or a configuration change needs to be distributed to one or

more nodes in the cluster, the GPFS system must use external remote shell and remote copy commands, which are defined at the time of initial cluster creation. Many GPFS commands are essentially special-purpose versions of a distributed shell command such as dsh(CSM) or psh(XCAT), and there is also a distributed-shell tool called “mmdsh” which is supplied with the GPFS distribution.

Any remote shell and copy commands may be used, but they must support passwordless authentication as the root user to all nodes in the cluster, and they must support the standard rsh/rcp parameters. Passwordless authentication is mandatory because the remote shell and copy commands are executed within the GPFS command, and because it is possible to have thousands of nodes in a single GPFS cluster. In modern Linux cluster environments, ssh and scp are typically used, with passwordless authentication configured via ssh keys or through host-based RSA authentication. In AIX/CSM environments, the use of rsh and rcp over private, non-routable networks is the default mode of passwordless root access. OpenSSH and hostkey-based authentication is used in SDSC’s TeraGrid IA64 environment.

6.2 GPFS 2.3 Multi-Cluster Authentication and Control

In multi-cluster configurations, a GPFS cluster may export its filesystems to other GPFS clusters, allowing nodes in these clusters to mount the filesystems as if they were local GPFS filesystems, using NSD servers in the remote cluster to achieve parallel filesystem performance over the internet. However, in order to use a filesystem controlled by a remote cluster, nodes in the mounting cluster must be able to acquire and maintain the configuration data of the remote cluster, such as the list of primary and secondary NSD servers for a remote filesystem. They must also be authenticated to those servers so filesystem data is not publicly accessible.

In the initial implementation of Multi-clustering in GPFS 2.3 development, the use of remote-shell commands was extended to distribute this information to all nodes of all clusters. Since multi-clustering support is particularly useful for sharing filesystems across administrative domains, the requirement of passwordless authentication as the root user was problematic from a security standpoint. Multi-clustering is also useful for sharing filesystems across HPC platforms, where preferred remote shell commands may differ; in these cases special system configuration changes must be made to allow the same commands to be used on all nodes in all clusters. This issue is problematic from both an administrative and a security standpoint. In addition, the use of passwordless root access was the primary mechanism for authenticating a client cluster to a serving cluster, and filesystem

network traffic was always unencrypted, creating the possibility of data interception through packet-sniffing.

Based partly on the experience working with this initial implementation in the SDSC-IBM StorCloud Challenge for SC’04, in which both SLES8 IA64 clusters in multiple administrative domains and AIX/CSM Power 5 clusters were used, the authentication and configuration distribution mechanisms were changed for the GA release of GPFS 2.3. In the new implementation, there is no requirement for any remote shell access between, and there are more control options for authentication and filesystem traffic encryption.

The GPFS 2.3 GA release uses RSA public/private keypairs to securely authenticate each cluster when multi-clustering is used. In this implementation, out-of-band transfer of RSA keys by system administrators is required to establish trust between two clusters, and specific nodes are designated for communicating configuration data when connecting clusters. A new command, “mmauth,” is provided to manage the generation and importation of RSA keypairs, along with access control for external clusters and exported filesystems. A new configuration option, “cipherList,” is used to require RSA authentication when connecting clusters, and also to enable encryption of all filesystem traffic if desired. For the cluster that will mount an external filesystem, two commands are provided to define and establish trust with external clusters and filesystems, “mmremotecluster” and “mmremotefs.”

The process of establishing trust is as follows. First, the administrator of each cluster that will participate in a multicluster configuration creates an RSA keypair and enables the configuration option for RSA authentication or authentication plus filesystem traffic encryption. When creating keypairs and changing authentication options, all GPFS daemons in the cluster must be shutdown. The administrators of the clusters then exchange the RSA public key files via an out-of-band mechanism such as e-mail. For the cluster that will export the filesystem, the administrator uses the mmauth command to add the remote cluster to the list of external clusters allowed to communicate with the cluster, and the administrator of the importing cluster uses the mmremotecluster command to define the server cluster, including selecting a set of nodes which will be used for establishing authentication with the remote cluster, and the mmremotefs command to define the remote filesystem. A special device entry is added to each node in the importing cluster to represent the remote filesystem.

Once these steps are completed, the remote filesystem can be mounted using the standard UNIX mount command. When the mount command is issued, the GPFS daemons use the RSA keypairs to securely authenticate to one of the set of designated nodes on the serving cluster, which then distributes the information that the remote cluster has authenticated to all NSD server nodes in the

cluster. All standard GPFS filesystem query commands can then be used on the client cluster, for example to list disk status in the remote filesystem, or filesystem configuration statistics.

In the PTF 2 update to GPFS 2.3, an additional per-filesystem access control capability has been added to the multi-cluster configuration options. Using the `mmauth` command, the administrator of the exporting cluster can control, for each importing cluster, which filesystems can be mounted by the cluster and whether the filesystem can be mounted in read-only or read-write mode.

It is also important to note that the multi-cluster configuration as described above is not limited to simple client-server relationships between clusters. A single cluster can export to many other clusters, and in testing at SDSC a single cluster has successfully exported one filesystem to clusters at Argonne National Labs and NCSA over the Teragrid network, as well as an additional GPFS cluster within SDSC. RSA public keys are shared between the various mounting clusters, as nodes in various clusters may need to communicate with each other to negotiate file and byte-range locks. In addition, a cluster which imports a filesystem from another cluster could serve another local filesystem to the same cluster, so all GPFS filesystems within a multi-cluster environment could be shared across all clusters within that environment.

7. Other Efforts

There are other supercomputing Grid organizations with a significant effort in Global File Systems. One prominent participant is the DEISA¹⁹ organization. That grid is tightly coupled enough to unify the UID space among GFS participants, obviating the need to wait for a more general authentication approach. DEISA's core GPFS sites are shown in Fig. 12.

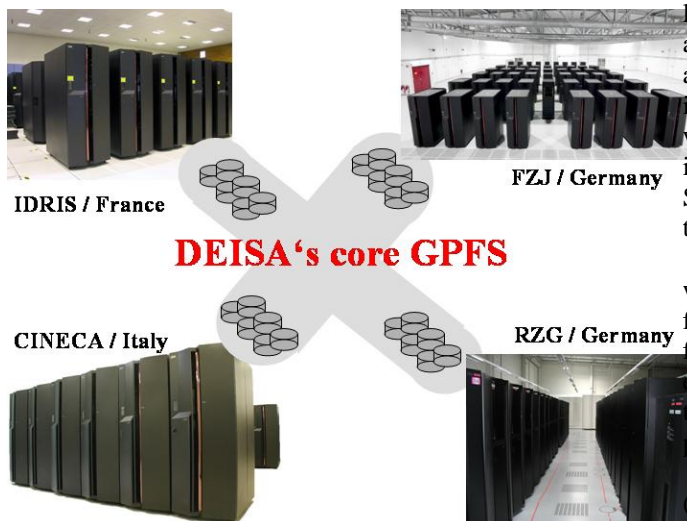


Figure 12. DEISA's core GPFS sites

Global file systems are the key technology for DEISA's data management. Among the four DEISA "core-sites", CINECA (Italy), FZJ (Germany), IDRIS (France) and RZG (Germany), IBM's Multi-Cluster (MC) Global Parallel File System (GPFS) has been set up, the world's first real production deployment of MC-GPFS. Each site provides its own GPFS file system which is exported to all the other sites as part of the common "global" file system. First functionality and I/O tests have been very promising and helped to discriminate between potential configuration options.

At present, the current wide area network bandwidth of 1Gb/s among the DEISA core sites can be fully exploited by the global file system. The only limiting factors left are the 1Gb/s network connection and disk I/O bandwidth.

This could be confirmed by several benchmarks, which showed I/O rates of more than 100 Mbytes/s, thus hitting the theoretical limit of the network connection.

Production-grade functionality and performance of the new global parallel file system could be successfully demonstrated with a real application. A resource demanding plasma physics turbulence simulation code was executed at the different core sites, using direct I/O to the MC-GPFS, the disks physically located hundreds of kilometers away from the compute nodes.

8. Future Work

It should be emphasized that the key contribution of this approach is not so much a performance level, or even a capability, but rather a paradigm. Disk price/performance ratios are one the very few parameters to be outperforming Moore's law in recent years, with disk capacity doubling roughly every 12 months for the last few years. This has lead to an explosion in disk availability and many more large datasets, both computed and observed, becoming commonplace. Such an increase in capability is not without its problems; few of the sites with (or considering) large disk farms have the necessary investment or expertise to run very large Hierarchical Storage Management Systems with automatic archiving to tape storage.

It is our contention that in the future many more sites will have large disk capabilities, but will tend to rely on fewer, centralized sites for data archiving, updates, verification and perhaps authentication. These will be the equivalent of "copyright libraries", which hold a guaranteed copy of a particular dataset, and from which replacements can be obtained after local catastrophes.

The Grid will be invaluable for replication of data. SDSC and the Pittsburgh Supercomputing Center are

already providing remote second copies for each other's archives) and Global File Systems will play their part as automatic caching becomes an integral piece of the overall file access mechanism. It seems likely that, at least in the supercomputing regime, data movement and access mechanisms will be the most important delivered capability of Grid computing, outweighing even the sharing or combination of compute resources.

In the specific instance of the SDSC GFS between now and next October we plan to:

- 1: Expand the disk capacity to a full Petabyte
- 2: Add another GbE connection to each IA64 server, increasing the aggregate bandwidth to 128 Gb/s.
- 3: Add a second Fibre Channel Host Bus Adapter to each IA64 server, allowing very rapid transfers from the disk to FC attached tape drives via a Hierarchical Storage Management System.

Eventually we would like the GFS disk to form an integral part of a HSM, with an automatic migration of unused data to tape, and the automatic recall of requested data from deeper archive. While the temptation is to allocate disk to users or application in a similar manner to the way computational resources are provided, data is a more long-term commitment and it is very hard to predict how long a particular data collection should stay on disk. In our view it is much more satisfactory to allow an automatic, algorithmic approach where data is migrated to tape storage as it is less used and recalled when needed. Obviously this depends on a significant investment in archival infrastructure and expertise, and is unlikely to be possible at all sites.

9. Acknowledgments

We would like to thank Larry McIntosh of Sun for coordinating the Sun effort in collaboration with SDSC at SC'02. We would like to thank Roger Haskin for coordinating the IBM support during SC'03 and SC'04. We would like to thank the SciNet organization for providing the significant bandwidth necessary to perform these demonstrations. The application groups for Enzo, SCEC, and NVO were also very helpful. A Global File System is of little interest without significant commitment from other sites, and we would like to thank Rob Pennington and his co-workers at NCSA and J.P. Navarro and his at ANL. We would especially like to thank NSF personnel for their interest and support in this approach. We would also like to thank Hermann Lederer for information on the DEISA GFS implementation in Europe. This work was funded in part by the National Science Foundation

10. References

-
- ¹ The Grid: Blueprint for a New Computing Infrastructure by Ian Foster (Editor), Carl Kesselman (Editor)
 - ² Foster, I., Kesselman, C., Nick, J.M. and Tuecke, S. Grid Services for Distributed Systems Integration. *IEEE Computer*, 35 (6). 37-46. 2002
 - ³ Catlett, C. The TeraGrid: A Primer, 2002. www.teragrid.org
 - ⁴ W. Allcock. GridFTP Protocol Specification, 2003. <http://www-fp.mcs.anl.gov/dsl/GridFTP-Protocol-RFC-Draft.pdf>.
 - ⁵ A.S. Szalay, The National Virtual Observatory, in ASP Conf. Ser., Vol. 238, Astronomical Data Analysis Software and Systems X, 2001.
 - ⁶ T. H. Jordan, C. Kesselman, et al., "The SCEC Community Modeling Environment—An Information Infrastructure for System-Level Earthquake Research. <http://www.scec.org/cme>
 - ⁷ Enzo – AMR Cosmological Simulation Code, cosmos.ucsd.edu/enzo
 - ⁸ Nimrod/G: An Architecture for a Resource Management and Scheduling System in a Global Computational Grid, Rajkumar Buyya, David Abramson, Jonathan Giddy, The Fourth International Conference on High-Performance Computing in the Asia-Pacific Region-Volume 1, 05 14 - 05, 2000, Beijing, China
 - ⁹ <http://www.supercomp.org/>
 - ¹⁰ <http://www.sdsc.edu>
 - ¹¹ A Practical Guide to Tivoli SANergy, C Brooks, R Henkhaus, U Rauch, D Thompson, IBM Redbook SG246146, June, 2001
 - ¹² <http://scinet.supercomp.org/>
 - ¹³ <http://www.nishansystems.com>
 - ¹⁴ GPFS: A Shared-Disk File System for Large Computing Clusters, Frank Schmuck and Roger Haskin, Conference Proceedings, FAST (Usenix) 2002
 - ¹⁵ State of the Art Linux Parallel File Systems: The 5th Linux Clusters Institute International Conference on Linux Clusters: The HPC Revolution 2004, Austin, TX, May 2004, P. Kovatch, M. Margo, P. Andrews and B. Banister.
 - ¹⁶ <http://www.ncsa.uiuc.edu>
 - ¹⁷ <http://www.sc-conference.org/sc2004/storcloud.html>
 - ¹⁸ A Security Architecture for Computational Grids I Foster, C Kesselman, G Tsudik, S Tuecke ACM Conference on Computer and Communications Security, 1998
 - ¹⁹ <http://www.deisa.org>