Design, Implementation, and Production Experiences of a Global Storage Grid

Phil Andrews, Chris Jordan,
San Diego Supercomputer Center
University of California, San Diego
Hermann Lederer,
Rechenzentrum Garching der Max-Planck-Gesellschaft
Max-Planck-Institut fuer Plasmaphysik
Garching GERMANY

andrews@sdsc.edu, ctjordan@sdsc.edu, lederer@rzg.mpg.de

Abstract

In 2005, the San Diego Supercomputer Center placed in production a large Global File System, consisting of over 500 TB of raw storage. Initial access to this resource was via the NSF TeraGrid, but this was later extended to non TeraGrid sites. In many cases, access rates to this centralized storage were faster than to local storage and authentication was handled by GSI certificates in a true Grid manner. Usage modes were both interesting and different from those anticipated, resulting in a major reconfiguration of the disk resource. Overall acceptance has been startling, with sustained daily growth rates in the 1-3 TB range. SDSC is working with IBM to closely integrate this GPFS file system with the HPSS mass storage system and to extend GPFS to include local caching. The intention is to provide an apparently unlimited capacity high performance Global Storage Grid for scientific researchers across the US.

1. Introduction

In early 2005, we reported at the IEEE mass storage meeting[1] on proof of principle work we had done using global file systems with both hardware assisted distribution and the native globalization of IBM's GPFS[2]. Later that year, we were able to put into production a large (500+ TB raw), high performance (7+ GB/s locally) file system that was exported to numerous sites across the US as the first nation-wide, high performance, global file system in production. Towards the end of 2005, we were able to also mount it at three European sites; spanning two continents and making more literal the "global" terminology.

In this paper, we will first describe the design of the file system, which emphasized performance, reliability and cost effectiveness by using the very latest hardware and software, including extensions to GPFS made in collaboration with IBM

Secondly, the initial usage by large scale distributed applications was examined: it proved to be significantly different from our expectations, with a large number of writes from production codes occurring. We discuss the reasons for this, and the modifications we made to the file system to accommodate this unexpected usage pattern. The final performance numbers are shown.

Thirdly, we describe the usage characteristics of the stable production system, both in overall growth and breakdown by application.

Finally we describe our plans to extend this into a truly Global Storage Grid, with the inclusion of automatic archival, replication, and local caching.

2. Design and Environment

The fertile ground for a large scale global file system and storage grid was laid by the high performance TeraGrid [3] wide area network.
At Supercomputing 2002, we demonstrated a Wide Area Global File System[4] within TeraGrid using FCIP encoding with specialized

hardware. At Supercomputing 2003 we used native GPFS[5] across TeraGrid without hardware assist for a global file system. At Supercomputing 2004 we showed a GPFS global file system at true Supercomputing performance levels[1]. All of these experiments provided us with experience towards creating a stable, high performance, production global file system.

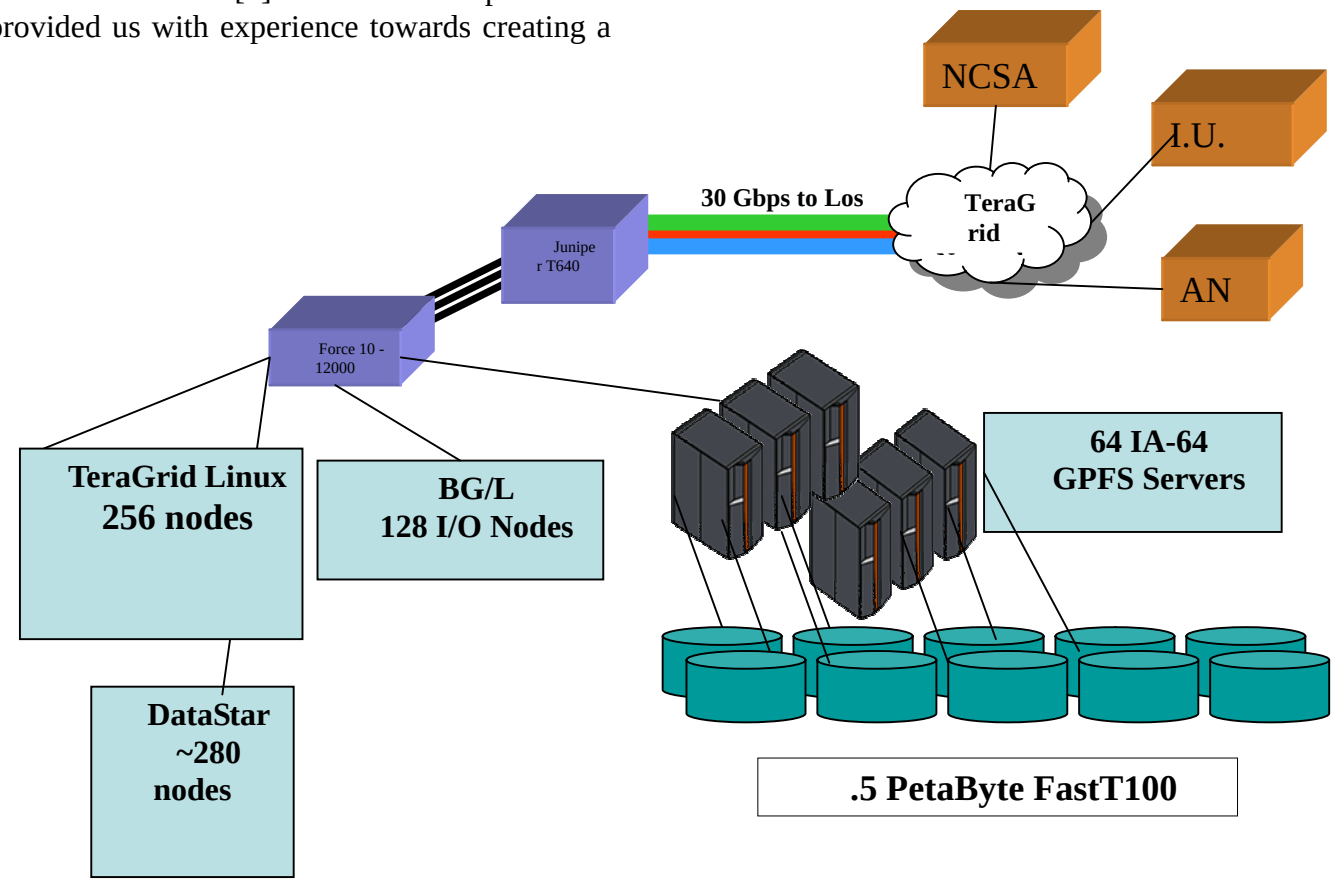

Figure 1. Global File System Across TeraGrid

In Figure 1., we show the situation of the GPFS global file system within TeraGrid. This was the initial configuration with the file system mounted remotely at NCSA, Indiana U., and Argonne National Laboratory. Since then it has been mounted at several other sites including Johns Hopkins University, Purdue University, NCAR, and the University of Texas. Approximately 0.5PB of IBM FastT100 (raw) disk is used for storage, with DS 4100 controllers and 250 GB SATA technology drives. Figure 2. shows in more detail 1/32 of the arrangement. Each half rack of disk technology consists of a dual controller DS 4100 unit with 67 250GB SATA drives. The servers are dual-processor Itanium systems with 4 GB of memory, a 2 Gb Fibre Channel Host Bus Adapter, and a fibre GbE network card. For redundancy, each disk subsystem is connected to two IA-64 servers, so we have a total of 32 disk subsystems and 64 IA 64 servers. Nominal maximum aggregate transfer rate is 8 GB/s to non-blocking Force10 GbE

switch. The drives were originally arranged in seven sets of 8+P RAID5 with 4 hot spares per disk subsystem.

Connectivity to the TeraGrid is via the Force10 12000 switch to the Juniper T640 router which connects to the 40 GbE TeraGrid backbone in Los Angeles through a 30 Gb/s link (since increased to 40 Gb/s).

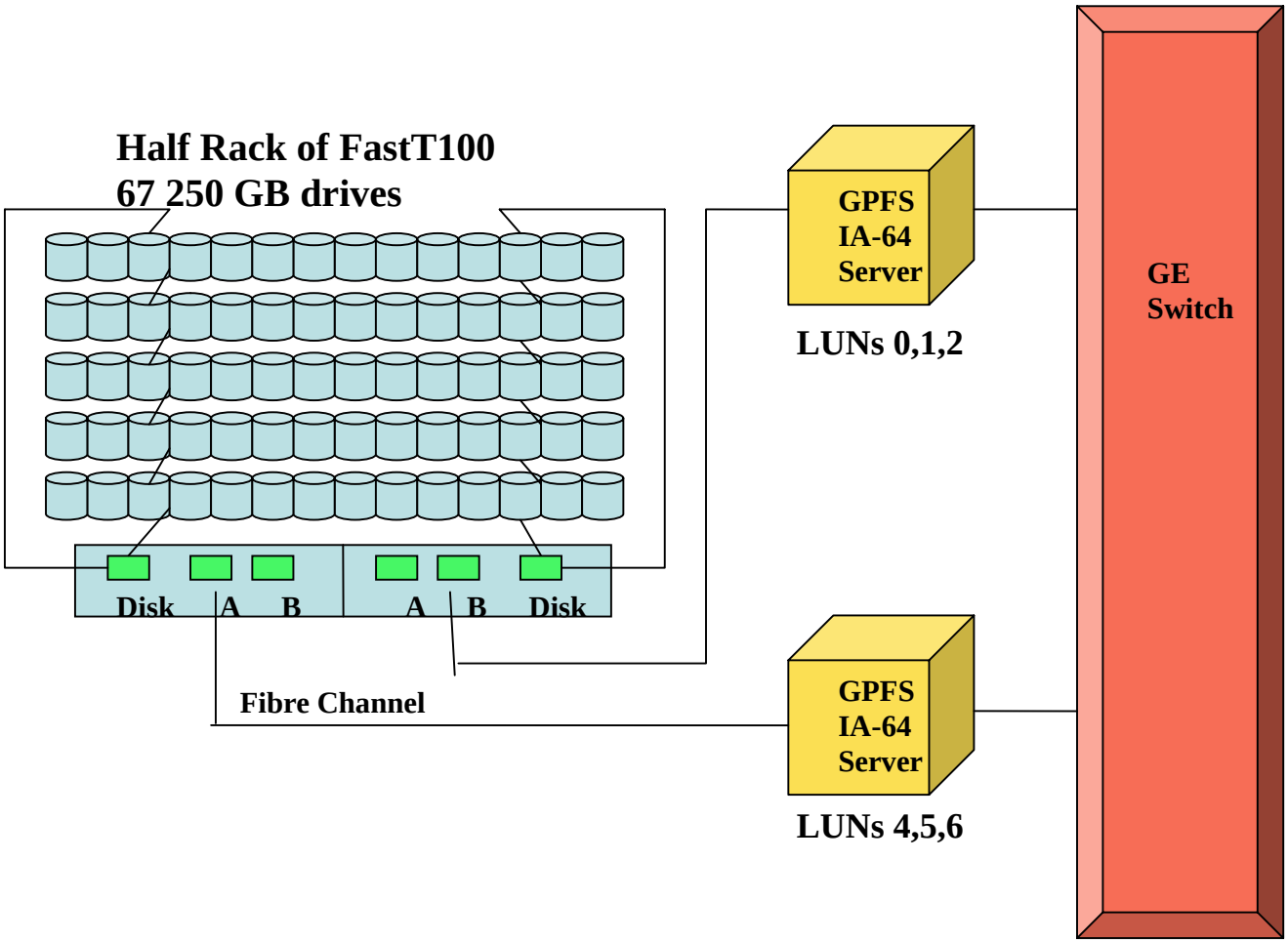

Figure 2. 1/32 of the disk and server arrangement

3. Implementation and initial experiences/performance measurements.

Initial performance experiments were to three sites: SDSC, NCSA, and ANL. The SDSC clients ran on three separate systems: a single Blue Gene/L rack, a 256 node IA-64 cluster, and a 280+ node Power4 system. The best connectivity was to the IA-64 cluster, as each of the 256 nodes have fibre GbE interconnect and a non-blocking connection to the Force10 switch, while the BG/L system has only 128 GbE connected I/O nodes while only a few of the Power4 nodes have GbE connectivity.

Initial usage was by the three sites at SDSC, a large (256+ node) IA-64 cluster at NCSA, and a smaller 32 node IA-65 cluster at ANL. In

August of 2005, friendly users were allowed access to the file system. Our original expectation was that the dominant mode of usage would be Read-Only access to very large, common datasets and the NVO (National Virtual Observatory)[6] dataset which exceeds 50 TB. The case for using Global File Systems to make available large, read-only datasets such as NVO is compelling: these are very important to many scientists, and in the absence of a central repository many copies would be stored at numerous sites. At over 50 terabytes each, this would lead to an enormous amount of duplicated stored data. Even more concerning, the problems of updates would be extremely daunting, with the specter of different extant versions creating enormous confusion throughout the pertinent scientific communities. Thus it was our expectation, that this would be the overwhelmingly dominant use of the of global file system, and in accordance with this assumption, we used a simple RAID5 setup with no provision for backups. Indeed, we expected that no unique data would ever be written to this file system, with observational or other datasets being moved there from archival storage systems. Optimization was purely for reads, with little attention paid to writes.

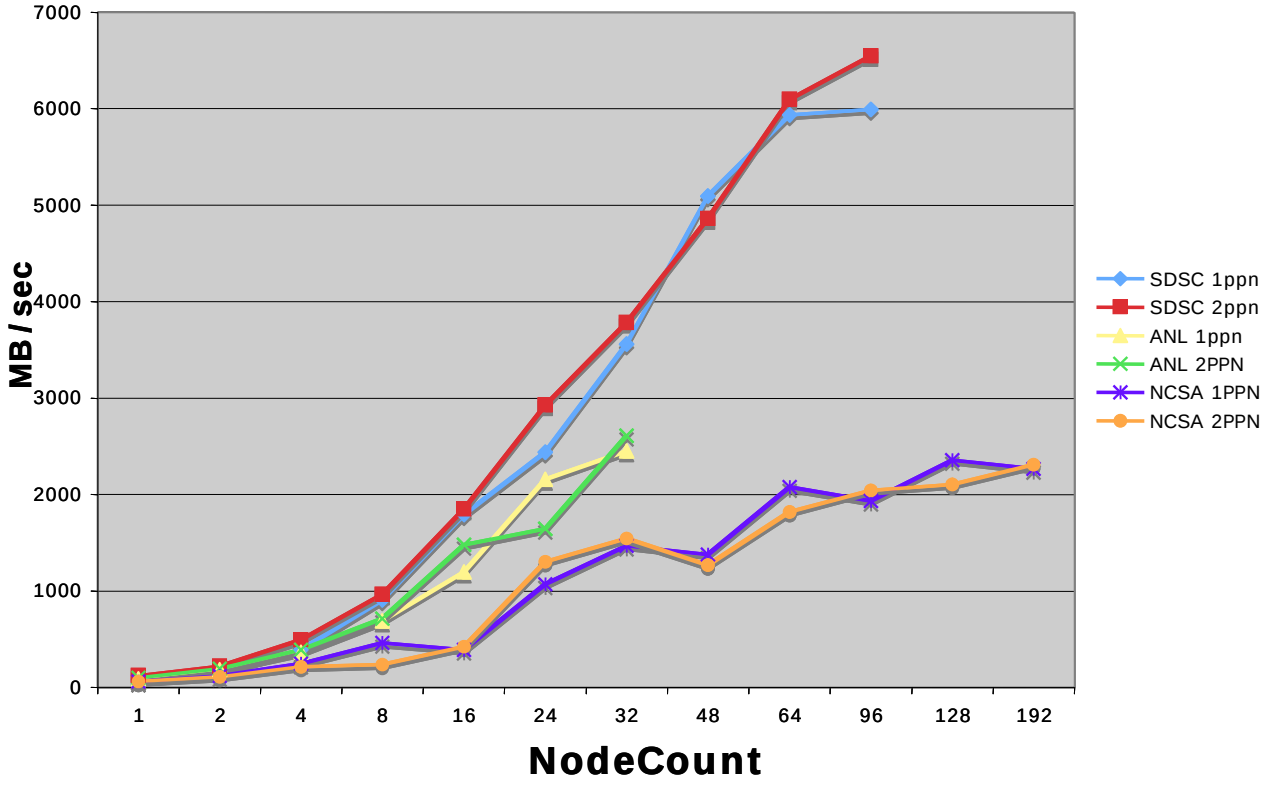

Figure 3. GPFS-WAN Read Performance in MB/S, 1 & 2 processes per node

During the friendly user period, however, it was decided to allow user writes (this had been a subject of debate) in order to give free rein to users to demonstrate how this new facility

could be used. To our surprise (and some consternation) a large part of the initial usage came as writes from running production codes.

The reason for concern was that the over 2,000 disk

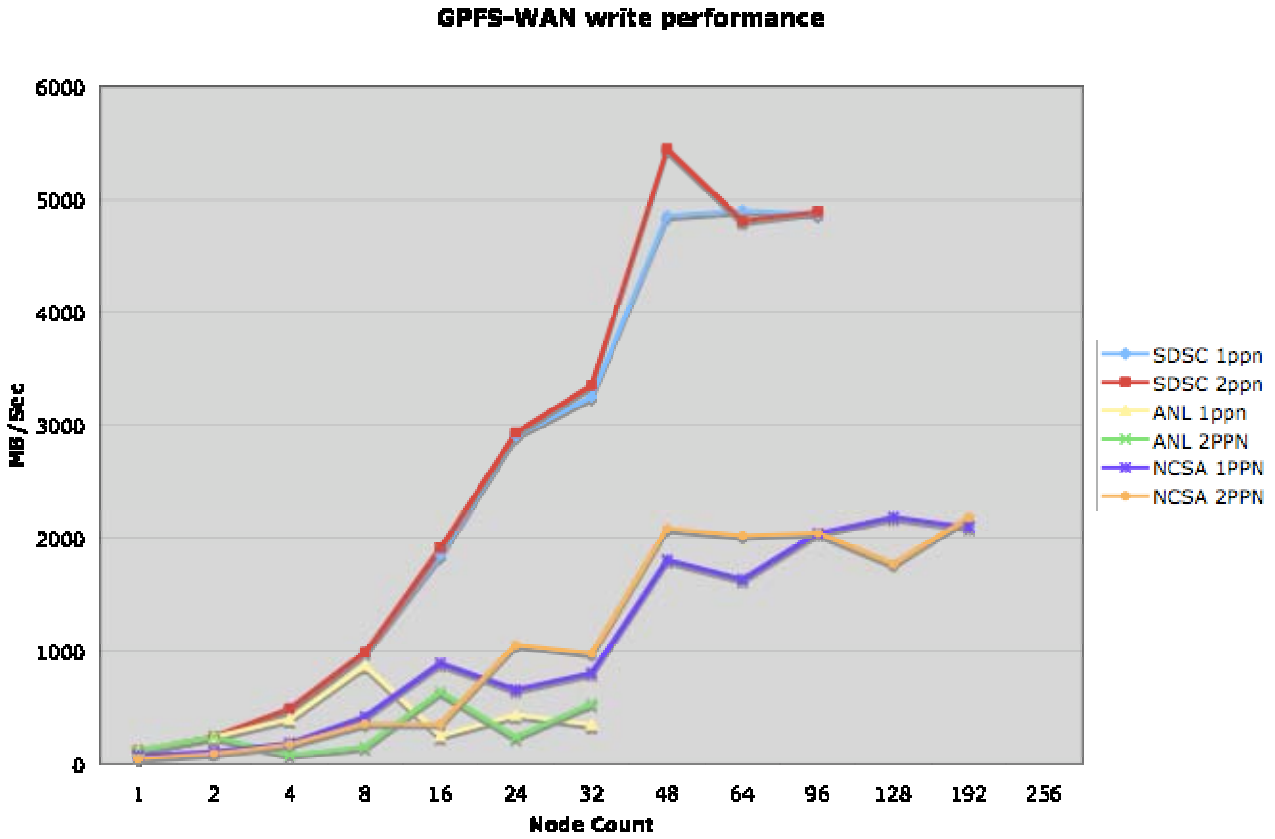

Figure 4. GPFS-WAN Write Performance in MB/S, 1 & 2 processes per node

spindles involved were rated for relatively low duty cycles, were arranged in a simple RAID5, 8+P, configuration, and were not backed up to an archival system. In any case, backing up 500 TB routinely would be both expensive and time consuming. Each disk subsystem contained seven 8+P Raid sets plus 4 hot spares. Total usable disk space was then 32 x 7 x 8 x 250 GB = 448 TB.

The user investment in data produced by supercomputer production codes is frequently enormous, and we were not comfortable with this situation. However, neither did we feel that we would be justified in restricting user behavior from what appeared to be a very attractive resource; usage was already increasing faster than we had anticipated.

Accordingly, we decided to drastically change the file system configuration, while exploring closer integration with mass storage systems. After some experimentation and discussion we decided to change the disk configuration from RAID5 to RAID10, i.e., striped plus mirrored. We availed ourselves of the hardware mirroring option on the DS4100 controllers, creating eight 4+4 RAID10 mirrored sets, plus 3 hot spare disks per subsystem. The total usable disk space was then 32 x 8 x 4 x 250 GB = 256 GB. By using hardware mirroring, we were able to both improve performance, and greatly increase the reliability as we could tolerate not only multiple drive failures but a single controller failure in each subsystem. This was at the cost of significantly reducing the available storage

capacity, but we decided that was a reasonable price.

After the disk rearrangement to what we expected to be the production configuration, we

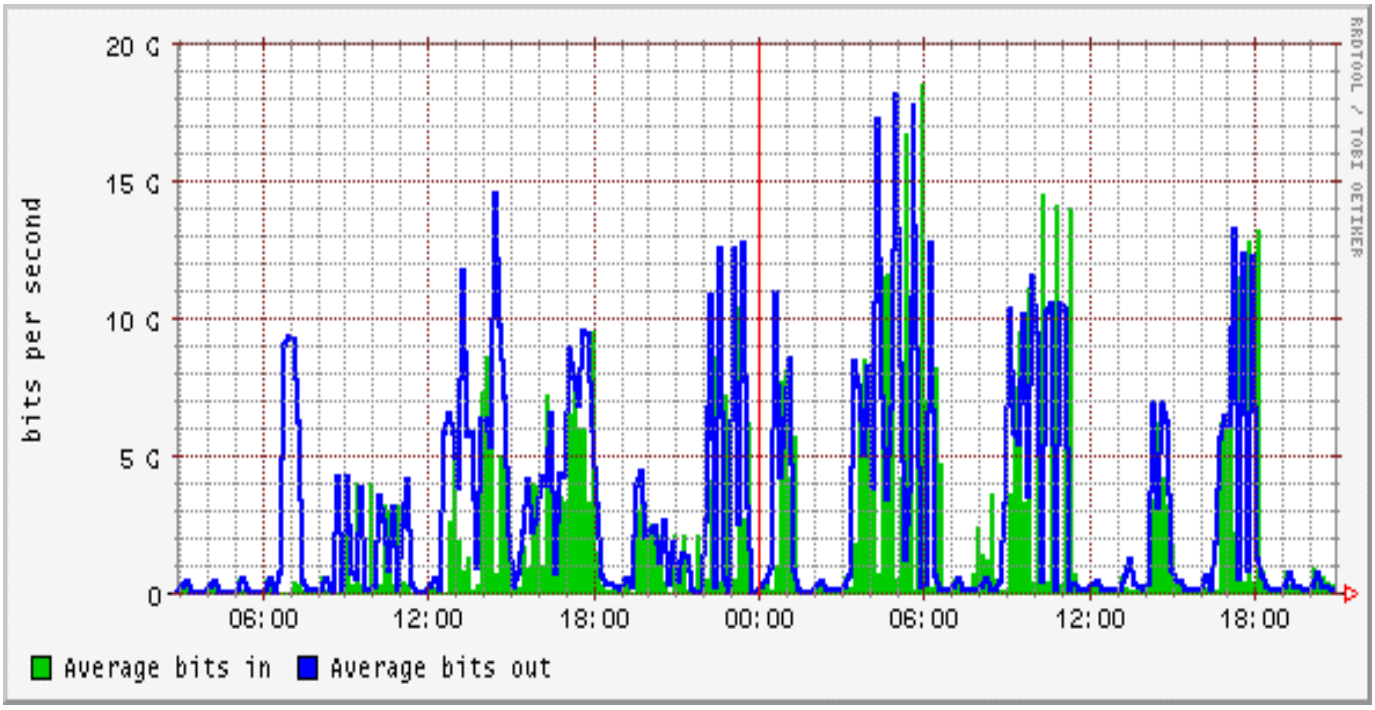

Figure 5. Networking traffic during file system accesses (5 minute averages)

took the chance to make performance tests from the file system to SDSC, NCSA, and ANL compute systems. In figure 3 we show the read performance in MB/s as a function of client node count to the 256 node cluster at SDSC, the 32 node cluster at ANL, and the 512+ node cluster at NCSA. These are all two-processor nodes, and the tests were run for both one and two processors per nodes, though differences between those two approaches were not generally significant. In each case, there was a single GbE connection to each node, although for SDSC the wide are network was not involved, while the other two sites required communication across the TeraGrid backbone. Thus, the maximum transfer rate to the SDSC system was limited by the GbE connectivity of the 64 IA-64 servers at 8 GB/s, while the other two sites were limited by the (then) 30 Gb/s connection between SDSC and the TeraGrid backbone in Los Angeles. Each of the sites could ramp up to its maximum at no more than 1 Gb/s per node. Given those limitations, the 6+ GB/s read rates to SDSC must be considered excellent, as are the 2+ GB/s reads across the network to ANL and SDSC. It is not known why ANL read rates rose faster than those at NCSA, but it must be noted that these jobs were not run on dedicated systems, and though each communicating node was used by the test job, other activities, both local I/O and network, were running on these systems.

Before the reconfiguration, write rates had severely lagged reads, but we had made several attempts to improve this in the new configuration, including using dedicated metadata servers, and in figure 4 we show the result of performance test writes from the same three systems as in figure 3. As mentioned before, these tests were not run on dedicated

systems, in this case the ANL cluster was very busy, leading to anomalously low results, particularly at high node count. Although the writes are still somewhat lower than reads, they are nevertheless very good.

While network statistics at the required resolution are not always stored, during the pre-production period we arranged to keep such statistics during a day when we expected reasonably heavy file system use. Unfortunately, this also coincided with a day which saw the TeraGrid backbone restricted to 20 Gb/s (two lambdas) by an industrial accident. Nevertheless, the network usage is shown in Figure 5, where peaks sometimes approach the 20 Gb/s maximum and often exceed the bandwidth (10 Gb/s) of one lambda.

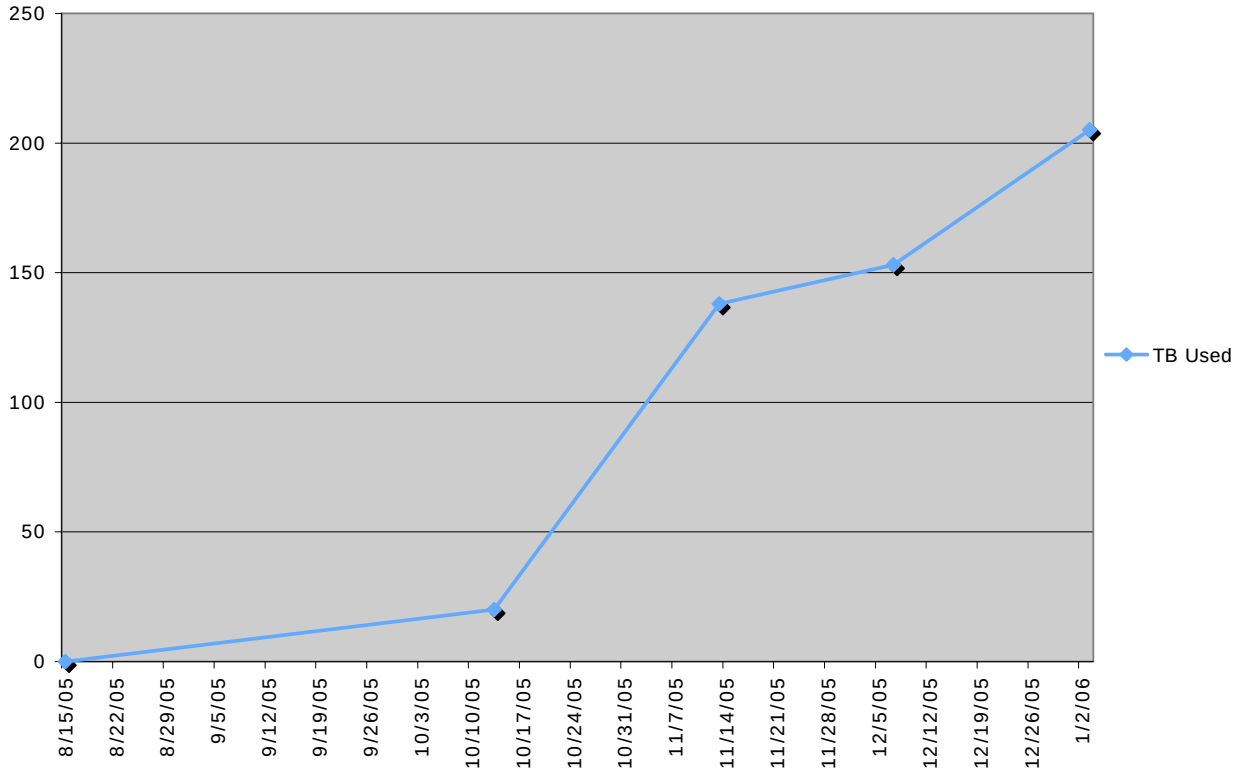

Figure 6. File System usage in Terabytes, 8/15/05-1/2/06

4. Production Usage

In early October, the file system was placed in production and received rapid acceptance by users. This was particularly pleasing, since for routine usage across multiple, the GSI certificate extensions that we built into the GPFS file system in association with IBM[1] had to be used. This was thus a truly grid application with GSI certificate authentication. To protect against unnecessary usage, write capability was not made routine with an explicit permission required. That also allowed us to keep some track of the data's origin. Presuming the correct permissions on the files, however, anyone could perform reads.

Growth in total storage immediately exceeded our expectations: in the period between the onset of production access and the Supercomputing ’05 conference, the daily increase was approximately 3 TB/day; comparable to the normal total growth in storage of the SDSC archival system! From SC’05 to the end of November, this slowed somewhat to approximately 1 TB/day, but then soon increased to 2 TB day through the end of ’05.

.

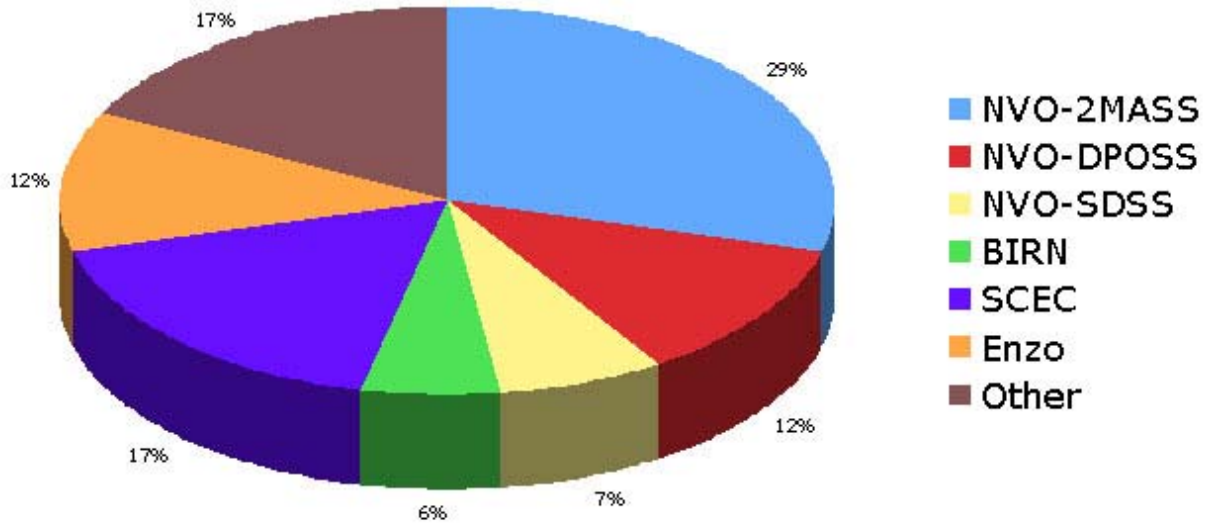

Figure 7. File system capacity usage by project

By the first week of January ’06, after approximately 3 months of production, the file system was over 90% full. Obviously it was essential that we understand the usage patterns leading to this growth rate.

As mentioned earlier, the original assumption was that the dominant usage patter would be accessing large, read-only datasets. Indeed, that could have been ensured by rendering the file system essentially read-only

with the only write access being from archived datasets. By allowing unimpeded writes, two very different usage patterns also arose.

In Figure 7, the current file system usage is broken down by project ownership. Over 1/3 of the total storage is indeed consumed by the NVO project[6] with its read-only dataset.

However, two other modes of operation are represented in Figure 7. In modern supercomputing, the investment in a particular code is often so large, that it essentially becomes a community project. Two of the projects in Figure 7 fall into that category. Enzo[7], and SCEC[8]. Although quite different in physical application; Enzo simulates galaxy formation while the Southern California Earthquake Center, as its name suggests, simulates earthquakes, there are some strong similarities in operation. In each case, a significant part of the process involves large dataset creation. In the case of both applications, this can be more than 50 TB. With these community codes, the actual dataset creation may be only 50% or less of the total computational effort. The data output is so large and complex, that a number of sites may be involved in its elucidation, including data mining and visualization. A common mode of operation would be a large run at a supercomputer site such as SDSC or NCSA, followed by the movement of data to other sites for extensive post processing. For these applications, the use of a global file system significantly simplifies and improves the speed of the post processing operations. Instead of moving the data in and out of archival systems and between participating sites, only one copy is required. There is even a significant advantage for the centers' archival systems; once the post processing is completed, the dataset can be deleted, generally in a matter of few months, without the requirement of backing up to tape. Thus, these applications seized on the global file system.

A third paradigm utilizes the global file system to allow pipelining through different resources. In the case of the BIRN[9] data the diversity is not in the various groups spread around the country each of whom would like to access the data, but in the resources that a single group of users would like to apply. In their case, the initial data is stored at SDSC while the computation is performed on the large NCSA Linux cluster with the output data written back to the global file system at SDSC from which it is visualized by a dedicated resource at Johns Hopkins University. Thus a complete three-site pipeline is utilized with no explicit data moves whatsoever. The researchers have reported approximately and order of magnitude increase in daily throughput using this method.

The remaining applications fall in the "other category". For some sites, the SDSC global file system is faster than their local file system, or may just be a convenient repository when local capacities have been exceeded. For others, it is a convenient means of communicating data across the grid. The convenience of having a certain set of files immediately available across numerous and diverse computing resources should not be underestimated. While explicit transfers via GridFTP or other protocols are always possible, the difficulties of keep rack of multiple versions, etc., can be daunting. Particularly for users who are fully concentrated on doing science rather than becoming immersed in the intricacies of operating systems and transfer utilities.

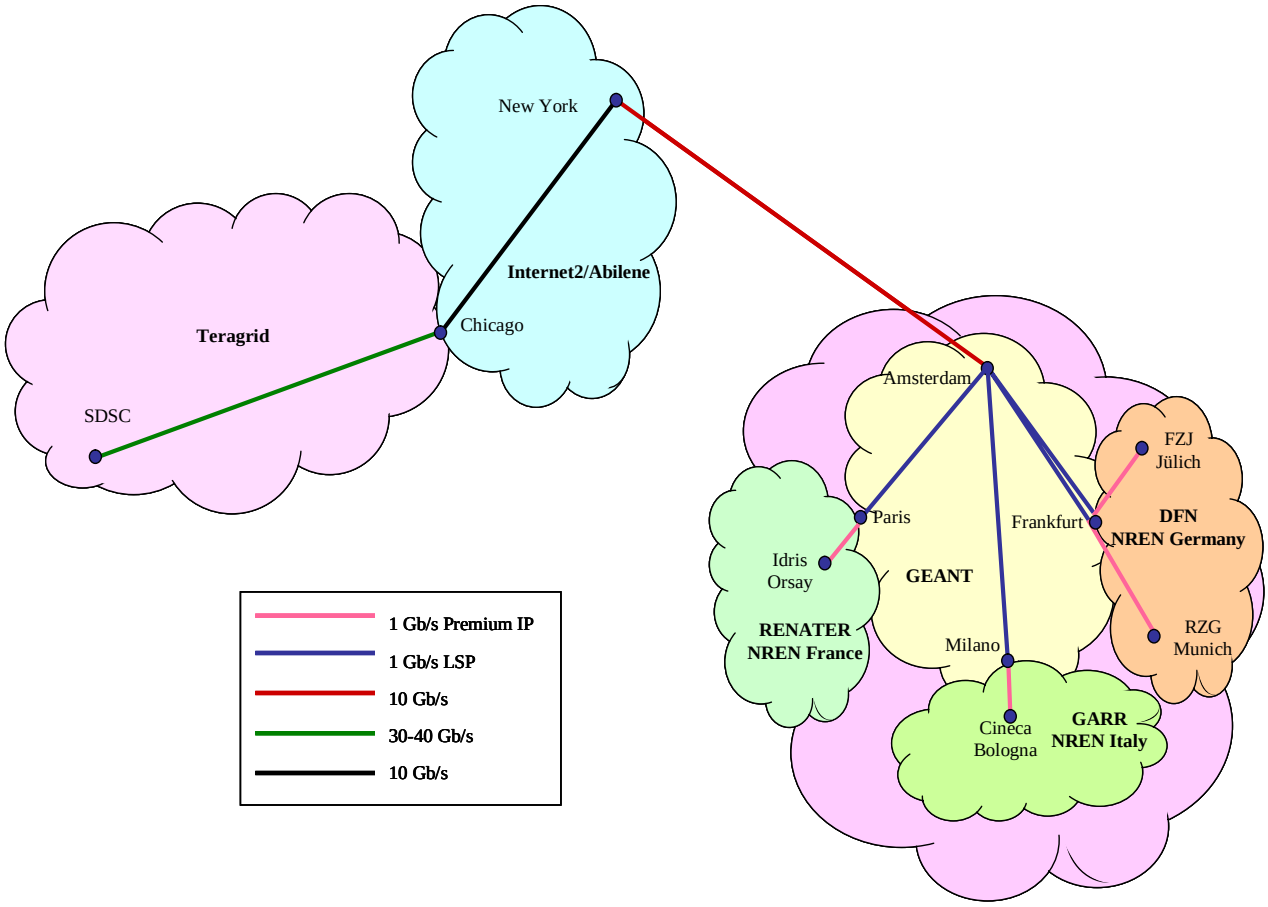

Figure 8. TeraGrid-DEISA interconnectivity diagram

4. Towards the true Global Storage Grid

While our experiences in the global file system arena have been gratifying, there significant steps that need to be taken for a completely self sustaining, and extensive system. In terms of the literal "Global" sense, one step was taken at the Supercomputing '05 conference, were cross mounting of the TeraGrid and DEISA[10] Global File Systems.

DEISA, the Distributed European Infrastructure for Supercomputing Applications, is an EU FP6 Research Infrastructure project. All major European supercomputing centers are jointly deploying and operating a unified supercomputing infrastructure on top of national services. The DEISA Consortium is constituted from eleven partners (BSC, CINECA, CSC, ECMWF, EPCC, FZJ, HLRS, IDRIS, LRZ, RZG and SARA) from seven European countries (Finland, France, Germany, Italy, The Netherlands, Spain and UK). The DEISA project started in 2004 and entered production mode in 2005.

DEISA and TeraGrid cooperate in several fields at various technical levels aiming at technological advancements useful for both initiatives, with a starting point and focus in the area of global file systems.

Efforts have been undertaken towards interoperability of both infrastructures, and first results could be demonstrated during SC05 at Seattle in Nov 2005 with a common, scalable, wide-area global file system spanning two continents.

A dedicated high performance network connection had been set up between four DEISA sites in France (IDRIS), Germany (RZG and FZJ) and Italy (CINECA) and four Teragrid Sites in San Diego (SDSC), Chicago (ANL and NCSA) and Bloomington (Indiana University).

The wide-area global file systems GPFS from IBM, used in production mode by both by DEISA (in an AIX environment) and TeraGrid (in a Linux environment) had been "cross-mounted" over the dedicated network, and a single high-performance global file system with a unique name space was created for scientists from the old and the new world.

While SDSC took the GPFS server role for TeraGrid with all disks physically located in one place (San Diego), the disk parts for DEISA were geographically distributed over France, Germany, and Italy with four sites with server roles.

Therefore, a cosmological simulation (ENZO) carried out at SDSC could transparently write its outputs to Europe, and even stripe it over France, Germany and Italy!
In another demo scenario, a gyrokinetic turbulence simulation (TORB) carried out at RZG, Germany, produced transparently output results in San Diego with online visualization of results in Seattle. Two more applications were also successfully used (another cosmological simulation, GADGET, and a protein structure prediction code (ROSETTA))

The joint DEISA Teragrid demo clearly demonstrated the growing importance of interoperability of grids at continental scope. Common data repositories with fast access, transparently accessible both by applications running anywhere in the grids, and by scientists working at any partner site as entry point to the grids, greatly facilitates cooperative scientific work at the continually increasing geographically distributed scientific communities.

In addition to geographical extent, we need to ensure the longevity of data in time. Obviously our current file system is almost full: we are working with both the GPFS and HPSS[11] developers to try to integrate our Global File System with a world-class archival system.

The intention is to make the Global File System at SDSC (or at least part of it) a visible front-end cache to our HPSS archival system. That would equivalence the reading and writing of files to getting and putting from a transparent archival system. Given that SDSC has over 50 tape drives available for archival storage, we hope that an acceptable transfer rate can be achieved even for tape accesses. In addition, we hope to extend the disk capacity in the short term, both to provide time before the GPFS-HPSS implementation is completed, and to improve the “hit rate” on disk resident files.

In parallel with this, we are working with the GPFS developers to try to improve local caching, so that latency is no issue, even for small files, and to provide another level of replication independent of that associated with the archival system. We have already installed an STK Silo at the Pittsburgh Supercomputing Center to create truly geographically distinct data backups in the case of necessary disaster recovery.

With the combination of large, high performance global file system, Hierarchical Storage Manager integration, local caching and remote backups, we feel that we are well on the way to a true Global Storage Grid.

There is further work necessary in several areas: an internal GPFS-HPSS integration instance is in place and being tested. With the extensive use of pfs-wan as a production facility, network outages that previously would have been unnoticed can cause disruption in a coincident I/O operation. An effort is being made to improve networking reliability, but it is also essential to improve the graceful exit of the file system in the case of an unavoidable network interruption; we are working with IBM to increase this capability

A more qualitative change is an attempt to provide lower latency figures and more robustness via local caching. Again, we are working with IBM in this area and hope to experiment with prototype software in the mid-term future. This should help significantly with the problem of small file accesses or small files, reducing the proportion of time taken by latency overhead issues. However, the difficulty of such an approach should not be underappreciated; presently there is no local caching, and the current implementation of replication, via failure groups, is synchronous and not designed for improving efficiency across wide area networks.

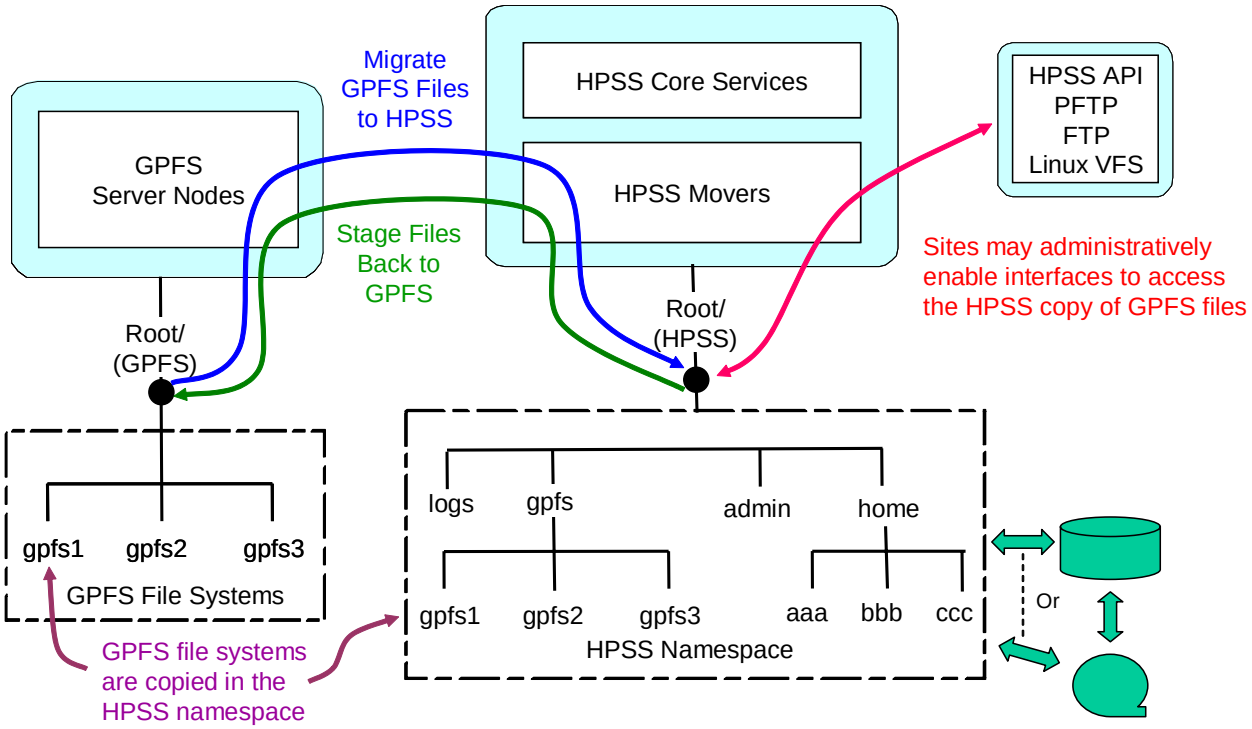

Figure 9. GPFS-HPSS integration diagram

We would like to thank the whole Enzo group, especially Robert Harkness and Mike Norman. Also, our colleagues at NCSA, ANL, and all of our other collaborators across the

Grid without whom this would have been impossible.

6. References

[1] Scaling a Global File System to the Greatest Possible Extent, Performance, Capacity, and Number of Users, Phil Andrews, Bryan Banister, Patricia Kovatch, and Roger Haskin. Twenty-Second IEEE Symposium on Mass Storage Systems, (April 2005)

[2] GPFS: A Shared-Disk File System for Large Computing Clusters, Frank Schmuck and Roger Haskin, Conference Proceedings, FAST (Usenix) 2002

[3] Catlett, C. The TeraGrid: A Primer, 2002. www.teragrid.org

[4] A Centralized Data Access Model for Grid Computing, Phil Andrews, Tom Sherwin, and Bryan Banister, Twentieth IEEE Symposium on Mass Storage Systems, (April 2003)

[5] High Performance Grid Computing via Distributed Data Access, The 2004 International Conference on Parallel and Distributed Processing Techniques and Applications, Las Vegas, NV, June 2004, P. L. Andrews, B. Banister, and P. Kovatch

[6] The National Virtual Observatory, Szalay, A. S. 2001, in ASP Conf. Ser., Vol. 238, Astronomical Data Analysis Software and Systems X, eds. F. R. Harnden, Jr., F. A. Primini, & H. E. Payne (San Francisco: ASP)

[7] http://cosmos.ucsd.edu/enzo/

[8] The SCEC Community Modeling Environment (SCEC/CME) - An Overview of its Architecture and Current Capabilities
Maechling, P. J.; Jordan, T. H.; Minster, B.; Moore, R.; Kesselman, C, American Geophysical Union, Fall Meeting 2004, abstract #SF41A-0754

[9] http://www.nbirn.net/

[10] http://www.deisa.org

[11] The parallel I/O architecture of the high-performance storage system (HPSS), R.W. Watson and R. A. Coyne, 14th IEEE Symposium on Mass Storage Systems