MARK URIS

DataMonster: building a cost-effective, high-speed, heterogeneous shared file system

Mark Uris has been doing system administration within the Scientific Computing Division of the National Center for Atmospheric Research for more than eighteen years. He is involved in the design and administration of infrastructure services for the Web, email, DNS, and security for NCAR.

uris@ucar.edu

IN THIS ARTICLE I DESCRIBE HOW WE built DataMonster, a high-speed heterogeneous shared file system. The project was built for the Scientific Computing Division (SCD) located at the National Center for Atmospheric Research (NCAR), where we support both supercomputers and very large datasets. The datasets are located on a multi-petabyte archival system made up of StorageTek's silos housed within SCD. The most popular datasets needed to be more readily accessible to NCAR's sixty-six-member university users conducting atmospheric and earth science research. DataMonster would provide the means to accomplish this. Over the course of this project, DataMonster's nature and composition changed, from a high-priced racing car to a cost-effective commuter car with race-car performance, by juggling choices for hardware and software.

The project named DataMonster called for a vast array of storage devices and heterogeneous servers interconnected by Fibre Channel (FC) switches and cables. It was needed to:

- Allow SGIs, Suns, IBMs, and Linux servers to share large datasets
- Eliminate supercomputers from countless accesses to the mass storage archives by having datasets online
- Supply the Visualization Lab with finer-grained datasets for visualization simulations
- Give users the ability to download datasets throughout the world
- Eliminate long wait times and high-priced processors for receiving data by using a hybrid of SATA- and FC-based storage units tailored for mid- to high-speed data accesses

DataMonster would outperform technologies such as NFS by replacing 25–30 megabytes/s maximum transfer rates with speeds in excess of 60–70 megabytes/s for writes and over 100 megabytes/s for reads. The initial amount of data targeted called for 16 terabytes, and estimates up to 100 terabytes were projected within a few years.

Experience with Storage Devices

We never lacked for benchmarks on storage device performance. A number of storage systems were herded into the NCAR data center for direct I/O testing against an SGI Origin 2000 including an Apple Xserve RAID, an Nexsan ATABoy, and a StorageTek ATA/SATA-based storage unit plus Sun and SGI offerings. SATA currently ran at a clock speed of 150 megabytes/s and ATA, also known as IDE, ran at 133 megabytes/s as outlined by their respective standards committees. Although SATA had a higher speed rating, it actually performed at comparable speeds to ATA. There was no rush among vendors to replace low-cost ATA disks with SATA ones. SATA-II was to be available in the immediate future and would represent a major speed breakthrough at 300 megabytes/s. The SATA standard committee had developed a ten-year road map where SATA would reach 600 megabytes/s and replace ATA technology. For more information on SATA see, in the June 2006 issue of *;login:*, Kurt Chan's article on comparing disk performance.

Although we hoped to discover a diamond in the rough, we found little variation from one system's performance to another's. For an ATA/SATA-based storage unit with FC, connection speeds of 50–70 megabytes/s were observed for writes and over 100 megabytes/s for reads. Units housing FC disk drives would double these speeds. The mainline vendors offered larger chassis capable of holding terabytes of disk, whereas vendors such as Apple had only a 3U chassis. Satisfied that we had a foothold on the storage subsystem of DataMonster, we turned our attention to finding a heterogeneous shared file system. This is where the journey really began.

Finding a Heterogeneous Shared File System

We had previous experience using SGI's CXFS file system in our Visualization Lab, but this was only on SGI servers. We know that CXFS ran on a number of different platforms, but we didn't know how it actually performed in a heterogeneous server environment.

We asked SGI to provide us with a list of reference sites running a mix of Sun, SGI, AIX, and Linux systems. Although the references checked out, using CXFS carried a hefty price tag. SGI controlled the entire hardware and software pipeline, from storage devices to switches. There is nothing like sticker shock to bring even the most lofty plans back down to earth. Estimates based on different storage configurations ranged from $18 to $20 per gigabyte for 20 terabytes of FC-based storage. This rate would decrease when SGI introduced SATA-based storage devices.

We started fishing around for alternative solutions. Sun had been doing a lot of work on their QFS shared file system. We studied it to see how it compared to SGI's CXFS. Sun didn't provide shared file service to all the diverse platforms in our environment. This was a show-stopper. Sun, like SGI, required use of their hardware (e.g., storage, switches) to use the product. The Sun quote for FC-based storage ran around $15 per gigabyte for 20 terabytes. Other solutions on the market were directed at Linux server configurations only, such as Cluster File System (CFS) Lustre, RedHat GPS, Sistana GFS, and the IBM GPFS just released for Linux servers. We appeared to be headed for a high-priced solution or no solution at all.

A New Solution Emerges

It was by sheer luck that I discovered in a trade journal Apple's plan to introduce their Xsan system in the immediate future. Apple touted their

shared file system as one-fifth the cost of the traditional offerings. Apple's plan involved using their Apple Xserve RAID storage devices to drive the storage cost down. Based on just storage, the average cost would be around $3.25 per gigabyte for a ATA-based system with an FC interface. This didn't factor in the cost of server software, client software licensing, HBA cards for clients, cables, and a switch. But it was well below the cost of other solutions on the market.

The Xsan would be managed on a Apple system running Mac OS X. As I dug into the details of Apple's shared file system I found that it was a rebranded ADIC StorNext file system (SNFS). Unlike other vendors, ADIC offered only the server and client software needed to create a shared file system, leaving selection of storage devices, switches, metadata server hardware, and operating systems up to the implementer. The metadata management software had been ported to a number of different operating systems. Twenty terabytes of storage suddenly seemed financially obtainable, but who was ADIC?

The ADIC SNFS

I hadn't dealt with ADIC before, so I made contact with the local sales rep. ADIC had traditionally done large-scale data management systems and had acquired Mountain Gate and their CentraVision file system (CVFS). They had set up a team of fifteen engineers in the Denver area, only an hour's drive from our site, to work on CentraVision, renamed StorNext File System (SNFS). We decided to set up a testbed to evaluate ADIC's SNFS performance in a simulated work environment where different vendor operating systems were interacting. This should reveal any problems before installing it on production servers. ADIC agreed to provide on-site installation and back-end support for setting up this trial evaluation of their product. The ADIC engineering staff would work through any problems we encountered.

Assembly of Testbed Components

To set up the ADIC SNFS, a Storage Area Network (SAN) had to be built. The SAN is the basic building block on which any high-speed file system resides. In a SAN, connections between systems and storage are made through one or more FC switches. The systems, storage, adapters, cables, and switches make up a SAN; the cables and switches are referred to as FC fabric. The SAN component was built with the following components:

- Apple Xserve RAID, with 1 terabyte of data space
- QLogic FC switch, with eight ports on the switch supporting 1GB FC
- Netgear four-port 100BaseT Ethernet switch
- Sun E450 server, ADIC SNFS client
- Sun 280 server, serving as ADIC SNFS metadata server and an SNFS client

The initial layout and testing dealt with making sure there was connectivity among the different system components (see Figure 1). What appeared to be a straightforward configuration turned out to require a lot of tweaking with cables and switch settings before the servers were able to communicate with the storage device. The Apple Xserve RAID came with an Apple FC HBA card that we used in one of the Suns. This caused a number of problems. The device driver for the card was set up for direct communication between the Apple storage unit and the server in which the card resided. When the FC line was run into the switch, it would dominate the

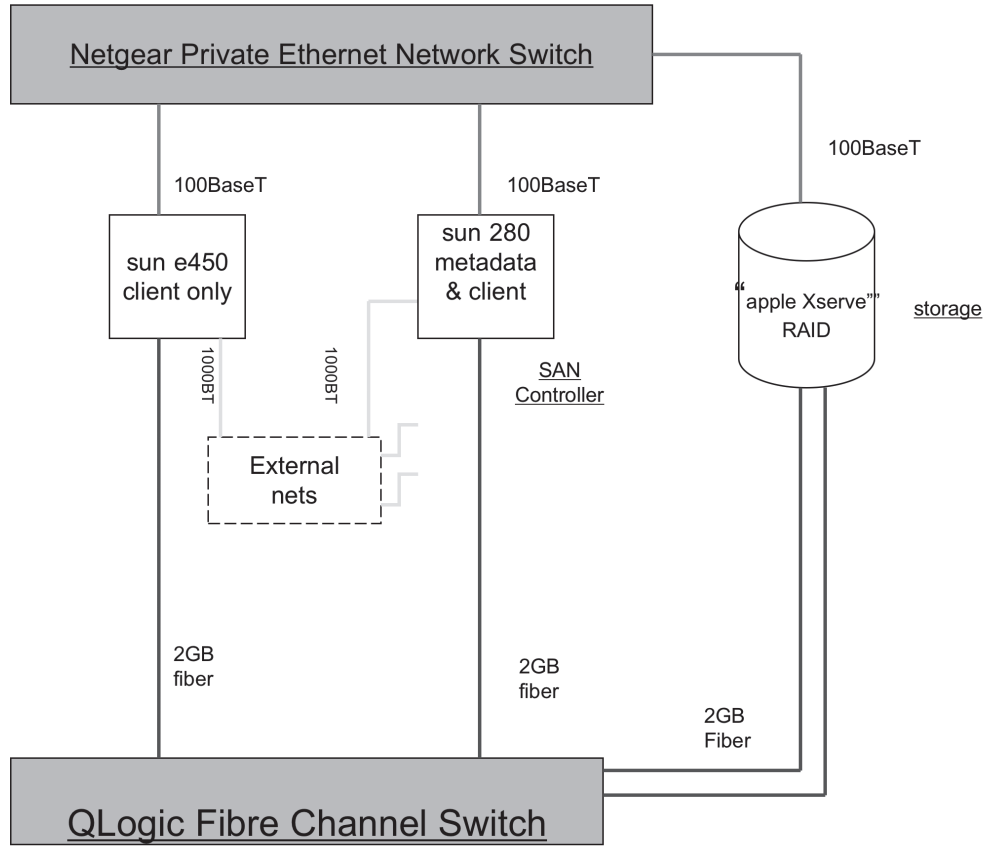

FIGURE 1: TESTBED.

communication channel and the other Sun server became unable to communicate with the storage unit. The problem was eventually overcome by buying another PCI FC HBA card from Sun. Once this was installed and run into the switch, both servers were able to communicate with the Xserve storage unit.

To secure the network between the servers and the storage device, a private Ethernet network was set up. This network was reachable only by logging in, using a security token, to the Sun 280, which housed the ADIC metadata server software. A Netgear switch on this private network allows the testbed hosts and storage device to communicate over 100BaseT Ethernet connections. It is isolated from the public Internet. The ADIC metaserver receives file requests on this private Ethernet network from the SNFS clients, but the actual file transfers are carried out on the SAN. SNFS requires dedicated space for metadata and journaling of the shared file systems that are created. This space should theoretically reside on a separate storage device in a production environment. But to expedite the installation of the testbed, one of the logical unit numbers (LUNs) on the Apple storage unit was used for this purpose.

In addition to carrying out metadata service, the Sun 280 server would be an SNFS client. The ADIC Web interface was brought up to do the actual file system creation and build. The SNFS configurations and builds were completed without any problems. We then installed the client software on each Sun server. It looked like we were headed into the home stretch. The only thing left was to mount the file systems on the clients, similar to what is done for NFS mounts. But the clients wouldn't mount the SNFS file system. A call was placed to ADIC to assist us in troubleshooting the problem. They were able to replicate the configuration we were running in their

lab, but there everything worked without any problems. An ADIC site engineer spent the entire day at our site inspecting the configuration. He couldn't find the problem either. Finally, one of the ADIC engineers asked what version of firmware we were running on the Apple storage unit. We found that we were two revisions behind the system they had in their lab. After we upgraded the firmware, we were up and running.

Performance Results from the Testbed

From our experience with previous tests, we were not really concerned about file transfer speeds since the metadata server is more of a traffic cop, and once the transfer takes place it has speeds almost identical to the storage devices directly connected to a server. We had heard horror stories of metadata servers becoming overloaded and becoming a bottleneck for file system activity. We needed the file system to be able to handle at least ten simultaneous transactions at any given time without impacting performance. We also tested files to determine whether any corruption occurred during system activity. The main standard benchmark suite used was iozone, which can be downloaded from the site www.iozone.org. Other available tools include xdd from the University of Minnesota and bonnie (www.coker.com.au/bonnie++). The initial results of testing were as follows:

- After more than 10 terabytes of data written to and from the file system, no file corruption or system crashes occurred.
- Aggregate sustained write rates were 50 megabytes/s, lower than we expected. We didn't know whether this was limited by hardware or the LUN sizes we created.
- Aggregate sustained read rates were 100 megabytes/s, higher than we expected.
- Over 100 files could be opened per second without degradation of metaserver performance.

Without attempting to optimize data stripe size or further tweaking to increase performance, we found the overall performance and stability of the system to be close to what we needed.

On to a Production Environment

The initial testbed results were reported to the different organizational units within SCD. As soon as possible, the dataset developers wanted SNFS to be set up between the Data Support section's Sun V880 server and the UCAR Community Data Portal's Sun V880 for a number of dataset projects that they were initiating. The amount of shared storage was set at 16 terabytes and was to be increased to approximately 80 terabytes within a few years. The testbed would have to wait.

We had tested and placed a Nexsan ATABoy system into production and focused on its big brother, the ATABeast storage system, which holds 16.8 terabytes. Two factors were involved in our decision. First, we didn't want to stack up a number of smaller-capacity (3.5-terabyte) units, which would mean running two FC connections for each unit into the switch and a more complicated file system layout. ATABeast would require only two connections, as opposed to ten connections for Xerve RAIDs. Second, Apple was staying with ATA technology and would not switch to SATA-II disks in their newer units such as Nexsan. Nexsan ATABeast had a price point of $2.90 per gigabyte. So the first storage unit in DataMonster would be named after a beast. Somehow everything started fitting together. We decided to add it to the testbed, where we could test it and tweak its per-

formance before installing it on two production servers. An evaluation unit was brought in for testing. If it performed well, we had the option of buying it. These are the results of benchmarking the Nexsan ATABeast with ADIC's SNFS:

- Approximately 7 terabytes of data were written and read back and no data corruption was observed.
- A single running process sustained I/O rates for large files at 100 megabytes/s for writes and 180 megabytes/s for reads without tuning stripe breadths or adjusting buffer cache sizes.
- The aggregate write rate for large files with multiple concurrent processes on two hosts was a little over 120 megabytes/s, and the maximum sustained read rate for large files with multiple concurrent processes was over 220 megabytes/s.
- The rate of metadata operations (e.g., file opens, closes, etc.) was a little over 250 per second and occurred with a single process making the system calls. This rate did not scale when an additional process on the host made system calls at the same time.

ATABeast performance had blown away any fears and doubts we had about a larger storage unit being slow. We had never seen numbers in this range for storage testing of ATA/SATA devices, let alone for a storage unit with a large storage capacity. This made the decision to go with a Nexsan ATABeast a foregone conclusion. Forget additional testing: We were going directly to production. All that was left was installing ADIC's SNFS on the Sun V880s. We used the Sun 280 from the testbed as the metadata server. Since it was already configured, no additional installation would be needed. We also needed a storage unit for housing the journaling and metadata for each file system we created. For this we used the Apple Xserve RAID from the testbed. The testbed was being devoured by DataMonster.

We installed a private network among the metadata server, storage devices, and Sun servers using a more robust switch than what was used in the testbed. Our data center had a McData Switch with 128 ports, and we used this as the main FC hub. A group of ports on the switch were zoned off for shared file servers and storage units. By this point, we had the production environment up and running without encountering any major problems. Another ATABeast was ordered and put into production within a few weeks after the first was up and running. DataMonster was starting to come to life.

Current ADIC SNFS Environment

The current production ADIC SNFS environment has been modified over the past year (see Figure 2). We replaced the original metadata server with a high-availability configuration of two Sun 210 servers with 8GB of memory on each server. The file system journaling was spread out over the Apple Xserve RAID, but closer investigation of the unit revealed that each of their two controllers only supports half the disk drives. We had thought that each controller communicated with all the drives. Losing a controller on the unit would cause half the file systems to be lost. Additionally, replacing the controller on the Apple unit wasn't a simple swap, so downtime was required to reconfigure it back into production. The Apple Xserve RAID was replaced by a Cipricio storage unit that resolved both of these problems. The Netgear switch on the private Ethernet network between the servers and storage unit was replaced by a Cisco network switch.

Another Sun V880 was added to the configuration, but this was to be used only for large dataset computations. The earlier ATA disk drives we used weren't truly serial but had been modified to imitate serial access. SATA ca-

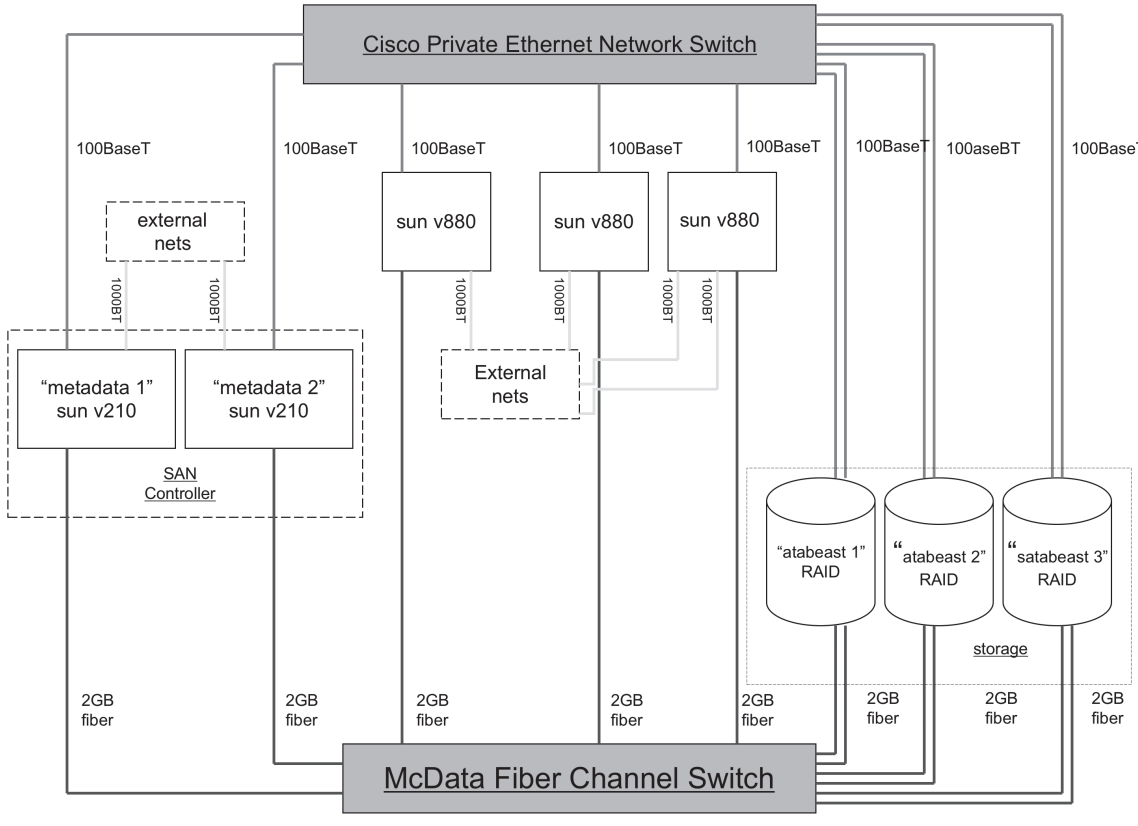

FIGURE 2: DATAMONSTER CURRENT CONFIGURATION.

bling works with ATA storage and peripheral devices. A new generation of SATA-II disk drives have emerged that are truly serial and run at double the access speeds of the first-generation ones in the earlier storage units. They run at about the same speeds as the FC disk drives. We then added a Nexsan SATABeast with these drives to the SAN; it was dedicated to shared datasets requiring heavy computational processing. The speed of the new drives ensured that I/O waiting would not cause a bottleneck for the processors.

Future Evolution

This is not the first nor will it be the last time we have had to transform a testbed into a production environment. After the initial pain of working through the testbed configuration (and I have heard similar war stories from other CXFS administrators), the production system is extremely stable and reliable. The next step will be to add SGI servers—used predominantly for visualization—to the ADIC SNFS system.

Unfortunately, ADIC has a restriction on the number of hosts it can support at 128. This is not a hard limit, but anything above this level would require major involvement by ADIC since it would saturate the metadata servers. This is a serious issue for us, because we run a large number of Linux servers. We are investigating the use of IBM's GPFS or CFS's Lustre for the large production Linux clusters. In the future we will probably run a hybrid of shared file systems, one for the heterogeneous servers such as Suns and SGIs and one for the large Linux clusters. User home directories and smaller static data would continue to reside on NAS systems. Still, we would prefer to run everything under one shared file system.

The overall cost of the system is a major consideration for us. The costs associated with using SNFS software are minimal compared to the hardware and software costs from a single vendor. The ability to select optimal storage devices and switches gives us a tremendous amount of flexibility in buying open-market cost-effective hardware. Based on previous experience and testing of shared file systems, we believe that ADIC's SNFS performed extremely well. The cost of ADIC's SNFS run around $5000/client, with no restrictions on the number of processors per client. Yearly maintenance runs around $5000 for 24/7 support. The amount to be spent on switches and servers is up to the implementer. The main benefit derived, of spending approximately $3 per gigabyte for storage, cannot be overemphasized, since there are no limits or restrictions on the amount of storage we can add to a system. In the future the cost of storage will be measured in terabytes, not gigabytes, owing to the boost in disk capacity per disk and the storage industry's tremendous growth.