Paper 197-2007

A Throughput-Intensive Compute and Storage Grid Using SAS® Grid Manager

Somantak Chanda, American Express, Phoenix, AZ

ABSTRACT

The time has come when a $2,000,000 SMP UNIX server can be replaced by a $200,000 grid of even greater performance, reliability, and fault tolerance. We'll discuss the considerations made in selecting the key grid components: grid node hardware, network infrastructure, and storage strategy. These must be optimized for the type of workload the grid will support.

We'll look at detailed performance metrics of our pilot grid, which executes single-threaded and massively parallel SAS workloads using SAS Grid Manager software on Linux. We compared numerous CPU, disk, and network configurations to optimize price versus performance. We'll discuss details down to the MB/s and how you can choose the right hardware for your processing needs, given your budget.

Finally, we'll discuss the business challenges of implementing a grid, including convincing your leadership of a grid's value, working with the IT Department, and software licensing.

INTRODUCTION

The International New Customer Acquisition group at American Express uses large databases of prospective customers to manage information-driven marketing campaigns to target 50 to 100 million people each year across several major countries. SAS is used on large SMP Unix servers, with a total user base of around 30 users, to process and store terabytes of data.

We started our journey to set up a SAS grid in the middle of 2006. The paper describes the factors taken into account and steps taken to design the grid that is expected to meet our growing data processing and storage needs while being cost effective and offering superior performance. Test results of our pilot grid and its key lessons will be analyzed. Meeting the challenges of convincing the leadership and other considerations are also examined. Our experience will hopefully encourage and provide insight to others who want to go the Grid way!

THE BUSINESS PROBLEM

We have two 6 year old Unix servers that are at capacity in terms of both CPU cycles and disk space, and can no longer fully support our needs.

The server used for processing related to campaign execution and other data mining activities like mathematical modeling and statistical analysis is a COMPAQ GS140 with 8 ALPHA CPUs (900 MHz), 12 GB RAM and 2.4 TB of disk. The server has 4 disk controllers. There are 4 SAS work directories, each of which is 177 GB. Performance of this server (real time taken to execute SAS programs) deteriorates drastically when we have more than four concurrent disk-intensive batch SAS jobs running. This is because when all jobs are accessing the work areas, dataset read/writes cause major disk thrashing. Occasionally the disk space of the work directories also falls short, when there are two or more programs processing large datasets in the same work disk. This causes both programs to crash.

The server used for executing other large production SAS jobs and ETL activities for a data warehouse is a COMPAQ GS160 with 16 ALPHA CPUs (900 MHz), 16 GB RAM and 3 TB Disk Space. One large parallel job to load the warehouse can take over the whole machine for more than 24 hours, making it unusable for other critical production processes.

WILL A GRID SOLVE THE PROBLEM?

Before we attempted to answer that question, we explored the possibility of meeting our needs by getting new SMP UNIX Servers.

EASY WAY OUT – NEW SMP UNIX SERVER

For leadership, especially those who are not technically savvy, the lowest risk, and therefore most attractive, choice is to replace the current systems with something similar. An SMP machine like SUN Fire E20K with 16 dual core CPUs (1.8 GHz) and 64 GB RAM could possibly meet our processing requirements and had a list price of about $1.4 Million.

A larger configuration of this server with 32 dual core CPUs and 144GB RAM had a list price of about $2.4 Million. This does not include disk storage. (Note that all prices mentioned in this paper are manufacturer list prices in September 2006.)

GRID CONSIDERATION – WORKLOAD PARALLELIZATION

Grid architecture works best when the data processing can be broken into multiple parallel branches. One of our needs is to run several single-threaded analytic and production jobs concurrently. The other is to run multithreaded jobs in batch mode that we break into pieces on our SMP machines. A grid is well suited to both of these processing needs.

GRID CONSIDERATION – SOFTWARE COMPATIBILITY

SAS supports both batch and interactive modes, and actively and passively supports grids. We can use automated grid scheduling capabilities in an easy-to-use, point-and-click interface. We also have the ability to develop traditional applications using the Base SAS programming language to run in a grid environment. SAS/CONNECT® provides the ability to distribute SAS jobs across the grid nodes, and must be licensed.

Our existing Base SAS applications can be easily grid enabled by wrapping them with few lines of pre-processing and post-processing codes. Users log into the grid manager machine, which appears to them as a large SMP machine with dozens of CPUs. A Linux script launches SAS and transparently submits batch jobs to grid nodes by automatically augmenting each job's code with the grid-enabling code lines.

We will use Red Hat Enterprise Linux (RHEL), as it is a proven grid OS, and is similar to our current Tru64 Unix system on the SMP machines.

GETTING STARTED

Initially we tried to identify the possible practical bottlenecks and technical unknowns in setting up our own grid. By creating a pilot grid, we could prove to our IT department and leadership that this new technology would work.

CHALLENGES

- The Sarbanes-Oxley act of 2002 is binding on our company and translates to higher level of IT regulations and compliance. Grid computing still being a new technology might pose additional challenges. The IT Department's approval is mandatory for our grid plans to succeed.
- We need to decide the exact hardware that will meet our performance needs, while being cost effective.
- We need to determine software licensing costs in a grid environment. The SAS pricing model is different for grids and can be significantly higher or lower per CPU core than a large SMP machine, based on factors such as total CPU cores, and selective product licensing (not all products need to be licensed on all grid nodes, so lesser-used products can be licensed on fewer than all nodes).

GOALS OF THE PILOT GRID PROJECT

The challenges we identified led us to define the following goals for the pilot grid:

- Prove that we can run our workload on a grid, with performance as good as or better than our SMP servers.
- Determine exactly what hardware and configurations (network, file server, etc.) to use for our production grid.
- Compare hardware and software costs on a grid versus new SMP hardware.
- Gain approval from our leadership and IT department to let us implement a production grid.

EARLY MOVER ADVANTAGE

Grid computing was in Gartner Inc.'s 2006 Top 10 Strategic Technologies list. Technology analysts predicted that this technology will be mature enough to offer value in the next 18 to 36 months, and that means it makes even more sense to consider it the right time to research – or adopt – soon.

We received enthusiastic support from many companies while conducting our pilot, due to its cutting edge nature. They contributed hundreds of thousands of dollars worth of professional services and loaned hardware and software for free to conduct our grid pilot.

CHOOSING HARDWARE

The nature of our workload was analyzed and the restrictions related to IT department and hosting requirements were taken into consideration in choosing hardware. The test results on the pilot grid helped us validate general performance considerations.

CHARACTERISTICS OF OUR WORKLOAD

A large proportion of our SAS programs is I/O intensive and involves heavy sequential reads and writes, especially to the work area. I/O bottlenecks cause major performance degradation for some of our processes. Such processes are a part of our production and ad hoc day-to-day computational needs.

Some of our large ETL processes utilize explicit multithreading and are CPU intensive. We have been using SAS version 9 for several years now, and internal multithreading of SAS procedures like PROC SORT and PROC MEANS also have better performance when more processing power is available. These procedures are a big part of the real time and CPU time taken by our processes. Calculation of mathematical and statistical formulas for our analytical processes will have better performance with faster processors.

Most of our current processes will work fine with 1 or 2 GB of RAM. Though the use of memory intensive programming techniques for few specialized processes may do better with more RAM, it is not crucial for overall performance of most of our needs.

Our servers are used by about 30 users, and we expect about 15 users accessing them concurrently on an average.

IT DEPARTMENT AND HOSTING REQUIREMENTS

The IT department regulations allow only certain equipment in our data centers. We need special permission for new or nonstandard items. This involves going through extensive evaluation and justification procedures. Large organizations often intentionally use non-cutting edge hardware and processes, in an attempt for greater reliability, which can be at odds with a project like ours.

The IT department has preferred vendor relationships with specific vendors and this limits the manufacturers from whom we can purchase hardware and software. It does let us leverage and use the already negotiated enterprise level pricing and relationship that Amex has with the vendors.

The maximum allowable server density per rack depends on the electric power and cooling capabilities of a data center. Our data center currently allows a maximum of 16 dual-CPU servers per rack.

The 24x7 maintenance need of the grid, the system administration resource needs, back-up and recovery plans are other critical things considered in detail. Hosting the hardware in our own data center can help us leverage the infrastructure already available.

CANDIDATE MACHINES FOR OUR GRID NODES

The following configurations were considered as candidate computational nodes for the grid. We tested the performance of these machines in our pilot. We tested both a low-cost alternative (which we could buy more of) and a higher performance machine. The HP DL380G5 became available as we started our pilot as an update to the HP DL380G4, which is why we tested both of these.

HP DL320 G4 (1U)	$2,300
3.4 GHz 2MB Pentium D	1 GB RAM
2 x 80 GB SATA Disk	Dual Onboard GbE
HP DL380 G4 High Performance (2U)	$7,750
2 x 3.6 GHz 2MB Xeon (+$500 for 3.8 GHz)	4 GB RAM
6 x 72 GB 10K RPM SCSI Disk (+$700 for 15K)	Dual Onboard GbE
HP DL380 G5 High Performance (2U)	$10,000
2 x 3.0 GHz Dual Core 4MB 5160 Xeon	8 GB RAM (ours had 4GB due to an order mistake)
8 x 72 GB 10K RPM SAS (Serial-attached SCSI) Disk	Dual Onboard GbE

The HP DL380 has redundant power supplies and fans for greater reliability, which the low-cost alternative does not.

A note on blade servers: Blades have many advantages. 64 HP dual core dual-CPU blades fit in a 42U rack. That is 256 cores per rack. They use 90% fewer cables, and their CPUs have lower power consumption. We did not consider blades, as our data center doesn't allow that much density, and we do not want to pay for decreased size we don't need. Moreover they don't offer as many of the “free” internal disk drive bays we want to use for SAS work areas.

CHOOSING RAM

The price of memory is low for smaller sizes and increases drastically once you have to get large capacity DIMMs. An example for the HP DL360 G5 High Performance machine, which has 8 slots for DIMMs:

4 GB	8 x 512MB DIMMs	$800
8 GB	4 x 1GB DIMMs	$1,900
12 GB	6 x 2GB DIMMs	$3,000
16 GB	8 x 2GB DIMMs	$4,100
24 GB	6 x 4 GB DIMMs	$25,200

NETWORKING POSSIBILITIES

Large (50 GB) sequential reads and writes from U320 SCSI disks mean we hoped to be capable of 200+ MB/second for a single job. Gigabit Ethernet (GbE) is 1 billion bits per second = 125 million bytes per second (125 MB/s), bidirectional, minus communication overhead. Thus GbE will maybe not even support a single job reading from a file server over the network. The advantage of GbE is that we have 2 onboard GbE ports on all servers for “free”.

10 GbE (about 1 GB/s practical max throughput) was still priced in the clouds ($3,500 per network card, and over $1,000 per switch port).

We heard that Infiniband would get affordable later in the year, but it was still pricey and perhaps hard to work with?

NETWORKING DECISION

We decided to trunk GbE ports to create a faster pipe to each machine. We were told that this was possible. We bought 4-port GbE cards ($750 each) for each machine, and assumed that we could achieve throughput of up to almost 500 MB/s.

Our concerns were whether we could we really get 100+ MB/s on each GbE line, and whether trunking would scale well. We bought a 48 port GbE “managed” switch that we believed would support trunking.

STORAGE STRATEGY

We expected to have most of our I/O intensive process steps in the SAS work area on each of the nodes. Putting as many disks as possible in the nodes increases the read/write speed of sequential data processing in the work areas. The HP DL380G4 can have 6 internal disks. Adding an external SCSI 14 disk enclosure is $4,000 for the enclosure (without disks) and a controller. We decided to test 10K RPM versus 15K RPM drives to see if the extra price is worth the performance gain.

Our work is segmented by process or country. Each segment needs 1 or 2 TB. Four 300 GB 10K RPM disks hold almost 1 TB in a RAID 5 configuration. We hoped that they could process 150 - 200 MB/s. Our strategy was to have numerous smaller RAID sets as opposed to fewer larger ones, to minimize disk thrashing.

We decided to use file servers which would serve files over the network via NFS. To save the complexity and cost of a SAN, our storage was simply SCSI enclosures directly attached to the file server machines.

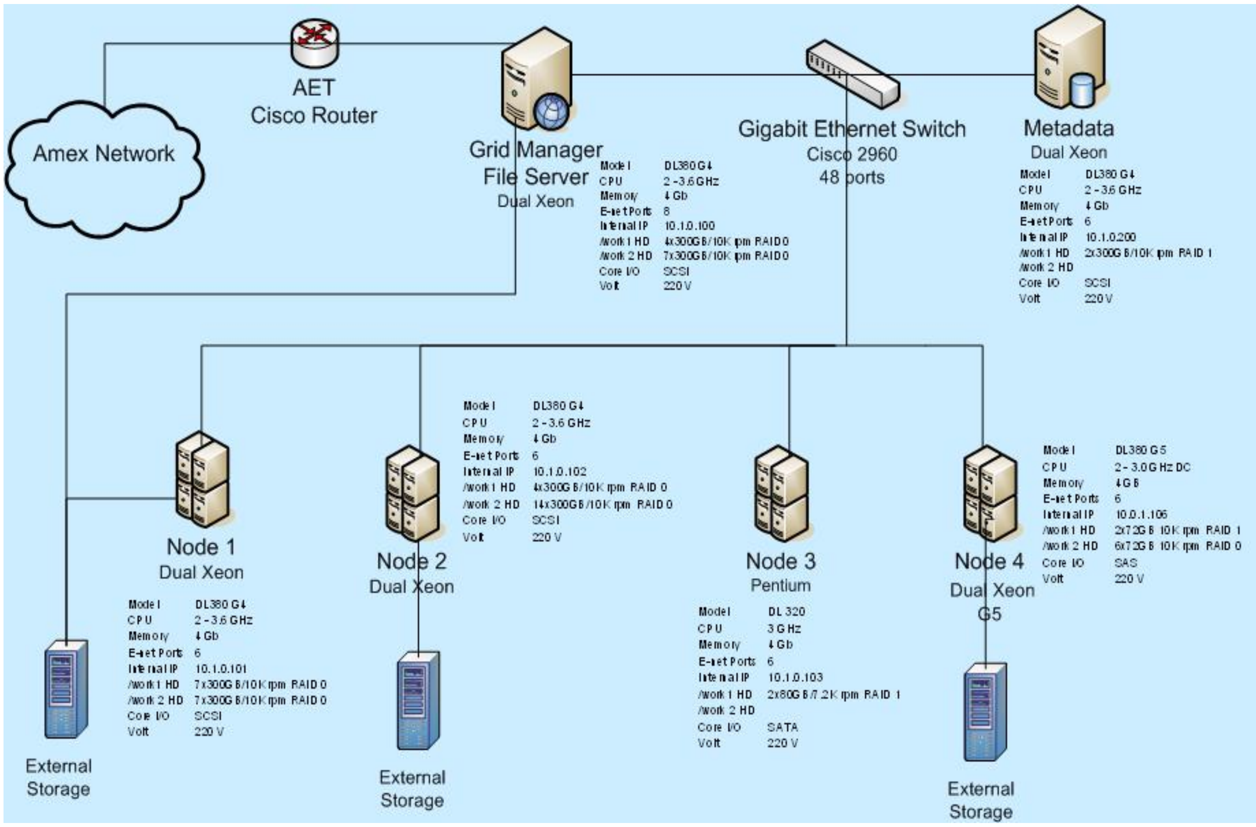

DIAGRAM 1: THE TEST GRID

SAS GRID MANAGER SOFTWARE

The SAS grid capabilities are made accessible to the user with a graphical user interface based product called SAS Grid Manager. SAS/CONNECT provides the syntax to enable distribution of SAS job components. The parallel processing capabilities of SAS/CONNECT have been integrated with components from Platform Computing to provide efficient workload distribution to the grid resources, efficient management of the grid resources, and run-time monitoring of the SAS grid environment.

One of the main components of SAS Grid Manager is the Platform Suite for SAS which has three components:

- Process Manager (previously Platform Job Scheduler for SAS) – serves as an interface to the SAS scheduling capability.
- Grid Management Services – communicates with the Grid Manager plug-in to provide run-time monitoring/management capability within SAS Management Console.
- LSF for SAS – provides mapping and load balancing of SAS processing across grid resources.

Some of the key aspects of using SAS Grid Manager are:

- Manages all grid workloads using SAS Management Console as the interface.
- Works in conjunction with Platform Computing LSF modules on each node.
- Grid configuration information and settings are stored in Metadata Server.
- Supports integration with other SAS products.
- All SAS jobs run initially on the grid manager machine, and SAS sends the job to a node.
- For jobs already made parallel using MP Connect, minor code adaptation grid-enables them.
- Interactive SAS sessions or Unix-script-parallelized jobs work fine on the grid.

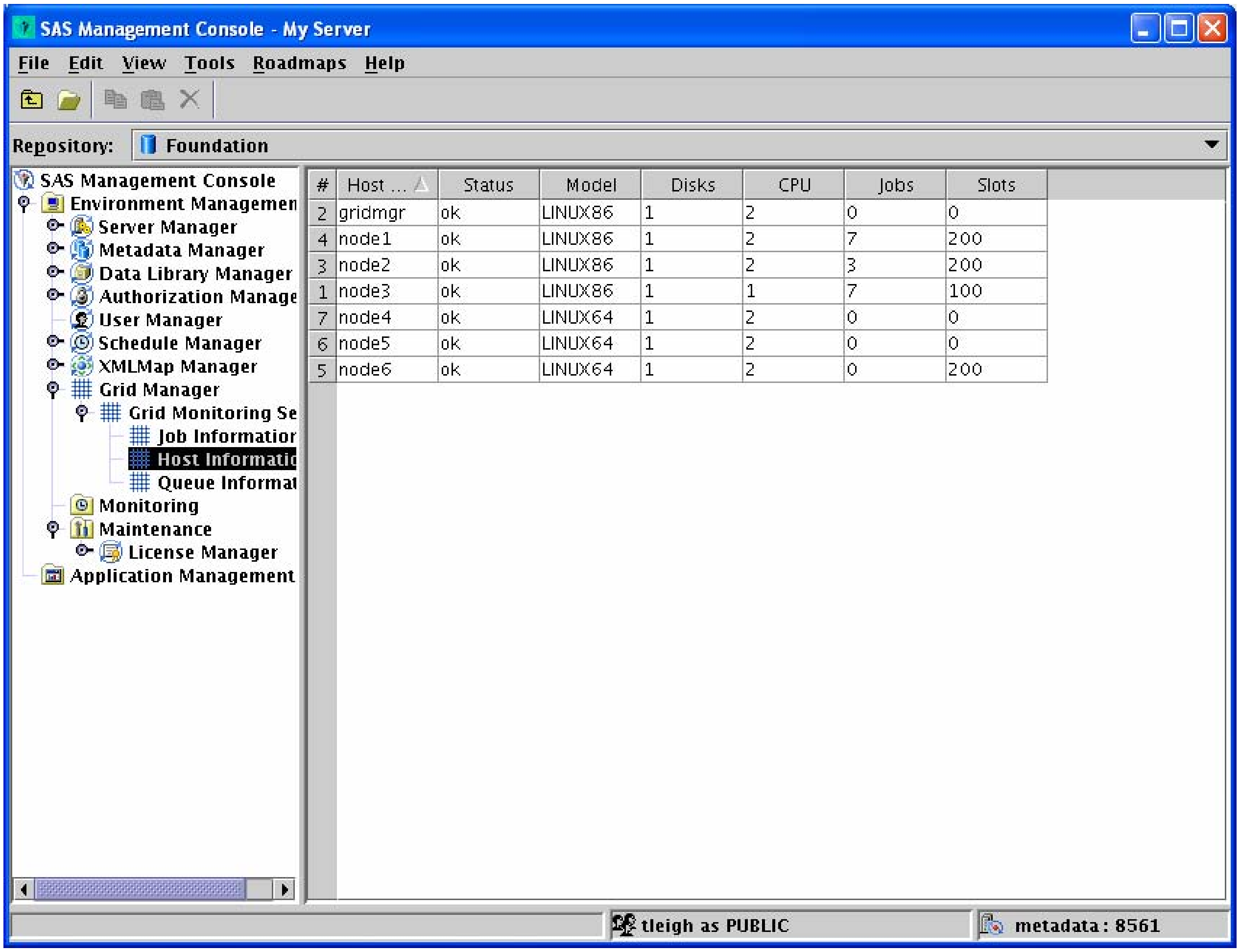

DIAGRAM 2: SAS MANAGEMENT CONSOLE

TEST DESCRIPTION AND RESULTS

We used the test grid to explore different hardware configurations. Certain relevant software settings were also experimented with. We used numerous different batch SAS processes, executed both separately, and multiple times concurrently on each hardware configuration. We had both Single-threaded and Multi-threaded processes tested. We had both specialized benchmarking jobs and actual production processes among the test programs. The benchmarking jobs were also run on our 8 x 900 MHz Alpha machine. This section summarizes some key tests.

SINGLE-THREADED TEST JOB

Most steps in this job use what we'll call a "test data set", a 24 million observation data set which is SAS Character-Compressed. It is 16.8 GB, 29% smaller than if not compressed. The job has 5 steps:

1. DATA _null_ step that reads the test data set (to test read performance).
2. Data step that reads the test data set and writes it to the Work library (to test read + write performance).
3. PROC SORT of test data set.
4. A CPU-intensive direct memory addressing routine that has negligible disk I/O (to test CPU performance).
5. Code from a recent production marketing campaign (normally run on our 8 x 900 MHz Alpha machine).

We executed this test job multiple times with different hardware configurations separately with no other processes sharing resources with the test job.

The table below shows the real execution time of the steps in minutes, on 3 different configurations of the test grid. It also has details of the test execution on our 8 x 900 MHz Alpha machine.

JOB STEP	8 x 900 MHz Alpha	3.4 GHz Pentium D	2 x 3.6 GHz Xeon	2 x 3.0 GHz Dual Core Xeon
Total	**167**	**198**	**65**	**46**
Read	10	3.5	3.5	3.5
Read + Write	23.3	28.2	7.6	4.7
SORT	53	123	18.9	13.6
CPU	27.9	10.1	8.4	5.6
Production	52.8	33.2	26.6	18.6
total time % of 8 x 900 MHz Alpha	**N/A**	**119%**	**39%**	**27%**

The Pentium D times are slow because SAS Work area was two RAID 1 SATA disks. The Xeon nodes had a 6 or 7 SCSI disk RAID 0 SAS Work area.

We executed multiple Concurrent Runs of the SORT Step on the 2 x 3.0 GHz Dual Core Xeon Node. The table below shows the real time of execution in minutes for all jobs to complete.

Concurrent Jobs	All Jobs Sharing 6 Disk RAID 0 Work Area	Each Job Has Its Own 2 Disk RAID 0 Work Area
1	13.6 (3.5% faster)	14.1
2	23.6 (22.3% slower)	19.3
3	32.0 (40.4% slower)	22.8
4	55.2	55.5

It had 10K RPM internal Serial-attached SCSI drives in the DL380 G5 for the concurrent run tests. Note how only using 2 disks doesn't really hurt us for one job, and steadily improves when multiple jobs are run. The node gets much slower after 3 jobs, indicating it is CPU bound.

16 THREAD PRODUCTION TEST JOB

This job reads a gzipped flat file and uses hashing to send its records to 16 subordinate SAS jobs. The 16 jobs create gzipped SAS datasets.

The reference system is a 16 x 900 MHz Alpha machine. Sorting is done using Syncsort. The job took 1 hour and 50 minutes.

Our test grid took advantage of only 3 nodes (5 CPUs): One 3.4 GHz Pentium D and four 3.6 GHz Xeons. Sorting was changed to SAS sort, and the main job directly piped its results to the sub-jobs instead of writing to disk. The job took 1 hour and 26 minutes.

ANOTHER 16 THREAD PRODUCTION TEST JOB

This job reads 16 gzipped flat files, plus supporting flat files. It does model scoring, prospect classification, and reporting.

The reference system is a 16 x 900 MHz Alpha machine. The job took 58 minutes. The test machine was just a single node (2 x 3.0 GHz Dual Core Xeon). Because this job dynamically assigns multiple SAS work areas, we attached a 14 disk SCSI disk enclosure and split it into 4 RAID 0 groups. The job was run unchanged, and it took 36 minutes.

This $20,000 of hardware outperformed the $1 Million Unix machine (five year old). Think about replacing your old 8 CPU machine with a single node!

INTERESTING RESULTS DURING TESTING

- You can't trunk more than one GbE line between any two machines, but a file server can trunk many together to serve files to numerous nodes.
- SAS read data over NFS about 35% faster than off of the same set of disks attached locally (writing is slower, though). This is apparently because NFS does a better job of queuing up data in large sequential reads.
- NFS can support 100+ MB/s data transfer over a single GbE line.
- When reading/writing from sets of RAID 0 disks using synthetic benchmarks, performance increased only up to 4 disks (about 150 – 200 MB/s). Adding disks beyond 4 had no real effect.
- 15K RPM SCSI disks performed about 15% better in synthetic benchmarks than 10K RPM SCSI.
- Presenting internal SAS disks to Linux as single disks (software RAID) performed better than using hardware RAID.
- Look for Full Bisectional Bandwidth in your switches. Our 48 port GbE switch had a backplane that only supported 6 Gbps, which would be a real bottleneck when using GbE for your grid's network.

PILOT LEARNINGS

The takeaways from the pilot helped us verify the suitability of grid architecture for our needs.

- A grid will work well for our workload. It will run jobs much faster than our old machines.
- The dual core Xeon G5 nodes give us the best single-job performance, and scale well to 3 concurrent very-resource-intensive jobs (sorts). Since our grid will often have a light load, the single job case will be most common for us.
- The 100 MB/s throughput of a single GbE line will be a bottleneck for concurrent jobs on a node if we want to have 3 jobs run concurrently per node each needing 100MB/s.

HOW USERS SEE THE GRID

To the end users the grid looks like one big Linux machine, as they only log into the Grid Manager machine. Users have an ID on each node for security, but they don't log into nodes. We will manage IDs centrally. A shell script invokes SAS and adds required code to grid-enable their programs. All storage is accessible from all nodes via NFS. All shared file system paths start with **/grid**, and file systems can be enlarged or moved to a different file server transparently.

GRID BENEFITS

FAULT TOLERANCE

The level of fault tolerance desired from the grid depends on its specific usage needs. Due to the distributed architecture, grids can intrinsically have high uptime as even if a node goes down, the design will ensure that the other nodes of the grid can share the load and continue to function seamlessly.

Our Unix servers crash every couple of months due to a bad CPU, fan, or memory. They may be down for 4 to 24 hours. The G5 grid nodes are server class machines. They have hot spare disks, hot spare memory, redundant fans and power supplies, and remote network console and power button access. They even have within six hour service calls from the manufacturer.

It may be good to keep spare parts on hand and keep a spare machine ready – just insert disks and attach any external storage and everything is back to normal quickly even if a node fails.

SECURITY AND ADMINISTRATION

We will use SSH (Secured Shell) to access the grid. Telnet, FTP, Xterm are insecure. Only the Grid Manager machine is connected to the outside network. Nodes and file servers are on a private network. All machine accounts can be maintained centrally via NIS or something similar. An image of each node can be made for consistency on grids with many nodes. Nodes can even boot over the network from a standard image.

SAS GRID LICENSING

The Pricing model was still evolving when we were working on the pilot. SAS licensing prefers 50 or more cores. As chip makers increase performance by adding cores, maybe the per-core model will change. Note that a single thread's performance doesn't improve if per-core performance remains static.

Grid allows licensing flexibility, so we can license rarely-used products on just one or two nodes and use job metadata to specify those nodes, when those products are needed.

Now that the hardware is so cheap, SAS costs more than the hardware. So if squeezing out extra performance is important; it may be worthwhile to invest a little extra in your hardware.

HOW TO CONVINCE LEADERSHIP TO ADOPT A GRID

It should now be a easy task, if we are able to back up sound rationale with even stronger cost saving figures:

- A grid is more reliable than a big SMP machine.
- A grid is just as secure.
- A grid can be extended and updated easily to meet changing needs.
- A grid with 36 Dual Core CPUs, 144 GB RAM, 7 TB SAS Work Disk Storage cost about $200,000, which is about 10% the price of a SMP machine like SUN Fire E20K (not including storage) with comparable configuration.

CONCLUSION

Our pilot was a great success. We proved that our workload would run great on a grid. Our IT department has allied with us and is considering making our design into a company-wide reference architecture. Our leadership chose to replace our old machines with a shiny new grid. We spent about $200,000 for grid hardware, compared to about $2 Million we'd have to spend for SMP hardware of the same processing power.

We purchased a 50 CPU core license from SAS, and are deploying 12 HP DL380G5 machines as the grid manager and 11 node machines. Each of these machines have 4 CPU cores. We found it optimal to allow 3 concurrent jobs to run on each machine, which leaves a CPU core for OS tasks and NFS processes.

We are using 2 CPU cores of the license to create a test environment, to use for low capacity SAS testing and testing of things like OS patches, per out IT department requirements. This test environment exists as two virtual machines using VMware software. This is great because you can set up a grid environment to test functionality without a large investment in hardware. (Obviously you will not see high performance in this configuration.)

Because the nature of the load on our grid is such that it is usually not being pushed to its maximum capacity, we found that having 11 high performance nodes is better than 44 lower performance nodes. Since SORT and MEANS procedures are naturally multithreaded, when a job is alone on a high performance node, there are multiple CPUs that can be leveraged to decrease execution time. Your organization's workload could possibly be handled better on a large number of lower performance nodes. Enough disk drive slots in the nodes to support a high speed SAS work area was something we found as the biggest limiting factor on the lower performance node. The Pentium 4 CPU was quite competitive with a single core Xeon, surprisingly.

While we believe that a GbE grid network could be largely sufficient, because it would limit us to about 100 MB/s for up to 3 concurrent jobs on each node, we decided to go with a higher speed network. We piloted a great new $300 per port Infiniband switch. Because Infiniband does not natively/easily support standard TCP/IP network traffic at speeds that much higher than GbE, and because our IT department doesn't want to support this "fringe" protocol, we are using 10 GbE. We're paying a lot for the network cards and the switches, but it is worth it to us to ensure that our network is not a bottleneck. The price for 10 GbE gear should consistently fall as it gains more popularity. The network cards have already come down in price by 50% in the last year.

While we can afford to have outages of up to a day or so in our environment, we never want them. To increase reliability, we ended up going with a clustered Serial-attached SCSI file system for our main storage, served by two file servers, as opposed to using direct-attached storage hanging off the file servers. If one file server goes down, the other file server can seamlessly take over. We were pleased to find that an entry-level SAN is more reasonably priced than we had thought.

We've eliminated the use of tapes for backups by buying a 30 TB SATA storage appliance and hooking it to a file server. A 30 TB SATA appliance can cost as little as $60,000, and provides surprising throughput performance.

REFERENCES

Doninger, C., and Wong, A. 2006, "SAS® Goes Grid – Managing the Workload across Your Enterprise," Proceedings of the Thirty-first Annual SAS Users Group International Conference (http://www2.sas.com/proceedings/sugi31/211-31.pdf)

ACKNOWLEDGMENTS

Thanks to Steve St. Peter, Director of Programming and Platforms at American Express, who architected our grid pilot and contributed information to this paper.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Somantak Chanda
American Express
2512 W Dunlap Avenue
Phoenix, AZ 85021
Phone: 602-537-8984
E-mail: somantak.c.chanda@aexp.com