Network Design Considerations for Grid Computing

Engineering Systems

How Bandwidth, Latency, and Packet Size Impact Grid Job Performance

by

Erik Burrows, Engineering Systems Analyst, Principal, Broadcom Corporation

Last Revised: June 8, 2012

Abstract

Network design is a key element in grid computing performance. Generally, a network with greater bandwidth and lower latency will result in greater grid job performance. However, the cost of any network design must be balanced with the resulting performance benefit. Understanding the impact of network design parameters on real-world job performance will help determine how much network equipment expenditure is appropriate for a new data center design or upgrade project.

Terminology

The terms **grid**, **cloud,** and **cluster** have less than precise definitions in today's computing industry. This document will use the term *grid* to refer to a set of *server-class* computers connected to a set of enterprise-class storage systems using an Ethernet network. The computers will be referred to as *compute servers,* and the storage systems will be referred to as *storage servers*.

The function of the compute servers is to execute units of work called **jobs**. These jobs are generally managed by a software system such as Load Sharing Facility (LSF) or Grid Engine. All grid jobs generate network load, in varying ways.

Job Types

For the purposes of this analysis, the following job types will be considered as the origin of network utilization:

- Electronic Design Automation (EDA) tools
- Software compilation
- Virtual machine (VM) live migration

The above job types generate network load in the form of the following operations:

- Network attached storage (NAS) data or metadata requests and responses
- VM image transfers

These operations can be broken down into the following more specific types:

- Streaming transfers: large NAS transfers and VM image transfers
- Small data transfers: NAS activity on small file sizes
- Small messages: TCP ACK packets and NAS metadata lookups and responses

The major variables in network design that impact network operation performance are:

- End-to-end network latency
- Bandwidth contention and oversubscription
- Maximum transfer unit (MTU) size

Optimizing most of these variables adds cost to any network design. This document will attempt to relate each variable to the performance of each of the above grid job types, enabling informed purchase decisions as performance relates to cost.

Network Load and Grid Job Types

This document will discuss three types of grid jobs and how they relate to network design:

- VM Migration
- Software Compilation
- EDA Tools

VM Migration

Virtual Machine (VM) migration is the act of transferring the running image of a virtual machine from one physical host to another. This form of network load is almost entirely streaming, meaning that the vast majority of the time spent performing this operation is copying data from the RAM of one physical host to the RAM of another physical host. The transfer usually does not involve any reading from or writing to a hard disk, so the action is limited by the CPU and memory speed of the servers involved and by network performance.

The frequency of VM migrations within a grid depends entirely on the job scheduler automatic migrations and the manual actions taken by system administrators.

The size of VM migration transfers is usually the RAM size of the virtual machine being migrated, typically several to tens of gigabytes.

Software Compilation

Software compilation jobs, as discussed here, are job types where a software project's source code is located on a storage server, and a job is run to read that source code, generate an output file, and store that result on a storage server. This document will not discuss methodology changes such as transferring source code to the local hard disk for this analysis.

A software compilation job usually runs hundreds or thousands of individual compilation tasks, usually with some level of parallelism. Each compilation task reads one or more typically very small files (1-10 kB), and writes one small file (10 kB – 1 MB).

EDA Tools

EDA tool jobs can span the range from streaming transfers to small file actions, and often include both.

The EDA process runs in stages, with each stage generating different types and numbers of jobs:

1. Verification
   - Many jobs, short run times
   - Software-like
2. Implementation
   - Fewer jobs, longer run times
   - Mix of streaming and software-like jobs
3. Fabrication
   - Few jobs, very long run times
   - Streaming operations, very large files

The real-world mix of job types in an EDA environment varies from one environment to another, especially the Fabrication stage, which may not occur at all in many EDA companies. In this analysis, **streaming write** tests are done to simulate large file accesses as with stages 2 and 3 above.

Network Performance As a Proportion of Job Performance

All grid jobs make use of network attached resources. Therefore, an increase in network performance will result in an increase in job performance. However, the job performance impact will depend on the amount of time that job spends waiting for network operations to complete.

If a job spends half of its run time reading and writing files to a file server, an increase in network performance can have a large impact, as long as the file server is not overloaded.

However, if a job spends only 1% of its run time waiting for network resources, very little impact will be seen in job performance regardless of how much improvement is made to network performance.

In testing for this document, the following table of job type I/O wait time ratios was compiled:

Table 1: Job Type I/O Wait Time Ratios

Job Type	I/O Wait Time Ratios
Software Compile	40%
VM Migration	90%
EDA Verification	0.6% - 3.6%
EDA Implementation	35% - 39%

Network Design Permutations

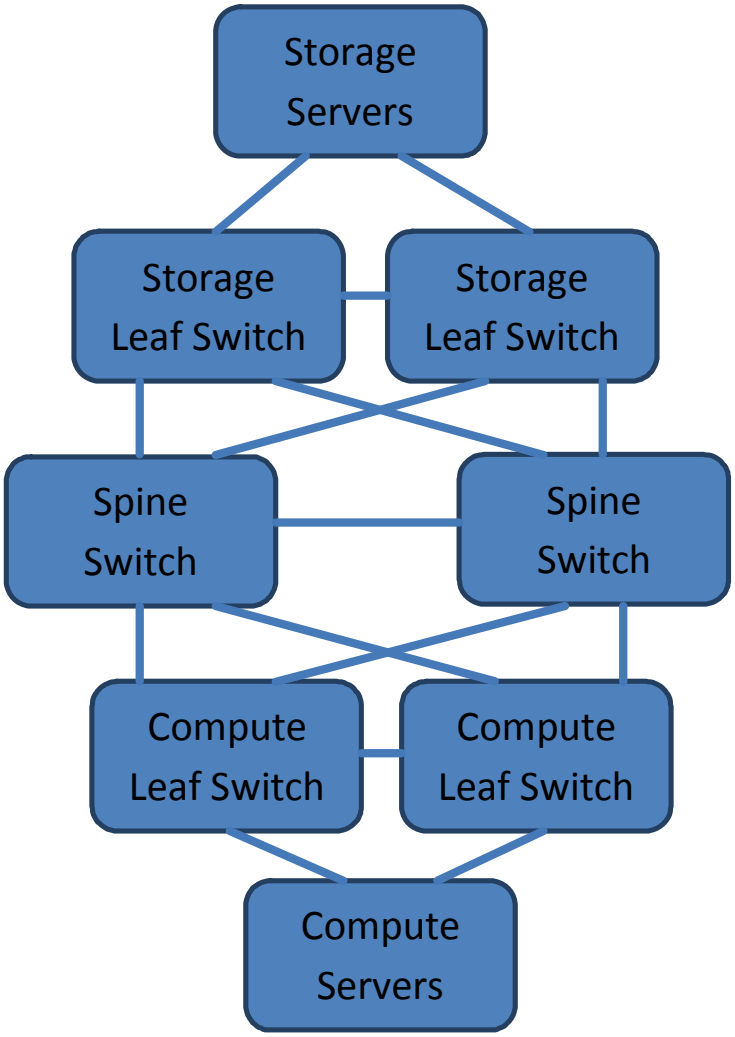

Figure 1. Elemental Network Diagram

Above is the most complete elemental diagram of an Ethernet compute-storage network. Most switch elements are optional, can be scaled, or duplicated:

- Either leaf switch layer could be eliminated given enough spine ports.
- Additional leaf switch layers could be added, to increase the number of storage/compute ports, at the cost of adding additional latency and oversubscription.
- The spine switch layer can be eliminated if enough leaf ports exist for a full-mesh interconnection.
- The leaf and spine layers scale horizontally for additional client ports, to the limit of the number of spine ports and leaf uplink ports.
- Storage and compute servers can be connected to one or more leaf switches, depending on the level of redundancy required and bandwidth needed.

Storage Bandwidth

Of the two end points, compute and storage, storage generally has the greater bandwidth requirement per-device. Determining how much bandwidth to allocate to a storage device can be complex if the cost of Network Interface Cards (NICs), cabling, and switch ports is a concern.

If the cost of storage connectivity is low compared to the cost of limiting the operating speed of the storage system, then determining connectivity is simpler: determine the maximum network throughput of the device and provide at least that much network connectivity. For modern high-performance storage systems, this will typically be an even-numbered quantity of 1-gigabit or 10-gigabit connections, split between two or more leaf switches for redundancy.

One way to determine the maximum network throughput of a storage device is to:

1. Create one file/LUN on the storage server, of a size no more than half the read cache of the device. This file will remain in cache, allowing the fastest possible read operations.
2. Connect the storage server to a large number of client machines using an Ethernet network similar to your intended production design.
3. Automate the operation of the client machines to re-read the file/LUN from the storage server repeatedly.
4. Eliminate any network bottlenecks.
5. Add client machines until the maximum throughput of the storage server is found.

The cost of under sizing your storage bandwidth is reduced job performance, and because one storage server generally serves data to hundreds or thousands of jobs simultaneously, that performance degradation is multiplicative.

Compute Bandwidth

The choice of compute server bandwidth is usually 1-gigabit Ethernet or 10-gigabit Ethernet. Testing done for this analysis shows that few single-thread grid job types exceed 1-gigabit/second in network load, except for VM migration and streaming file I/O.

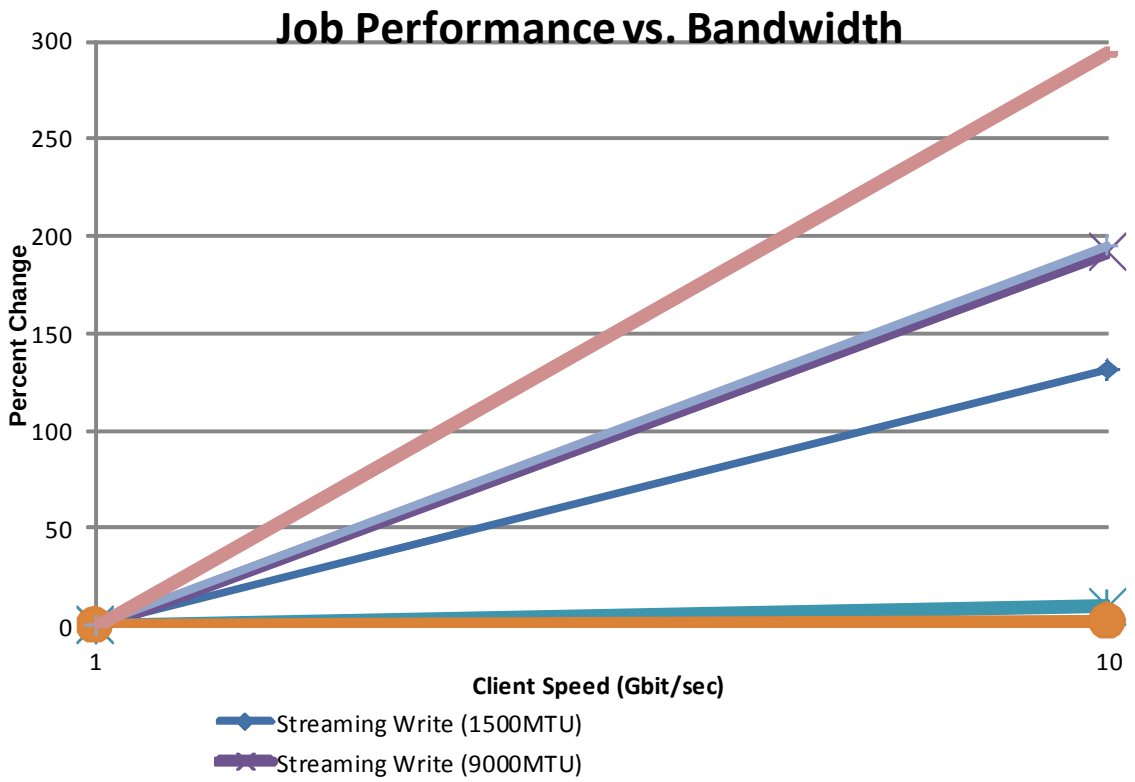

Figure 2. Job Performance vs. Bandwidth

In Figure 2, streaming write, non-parallel software compile, 8-thread software compile, and VM migration jobs are compared when run with a 1-gigabit and a 10-gigabit compute server NIC. The storage server has a 10-gigabit NIC for all tests. The difference between 1-gigabit and 10-gigabit is shown as percent change job performance improvement. The streaming write job showed a 131% improvement, and the VM migration job showed a 195% improvement when a 10-gigabit NIC was used.

Note that the VM migration test showed the CPU in both systems fully utilized, indicating that faster CPUs would yield greater performance. The CPU used in this test was an Intel® 3.4Ghz X5690.

Each job type consumes different quantities of network bandwidth. Each job type considered here is shown in Table 2. These values were the highest observed for each job type, which occurred using 10-gigabit Ethernet and a zero-hop network, with jumbo frames enabled.

Table 2: Job Type Maximum Bandwidth Utilization (10 Gb/s, 6-Meter Fiber Network)

Job Type	Client Transmit (Mb/s)	Client Receive (Mb/s)
1-Thread Compile	22.53	4.9
8-Thread Compile	148.6	32.4
VM Migration	5,121	13.6
Streaming Write	3,878	34.9

Bandwidth Multiplication

The bandwidth utilization figures shown in "Compute Bandwidth" on page 8 represent a single job only. In a production environment, each compute server typically runs several jobs, often as many jobs as the system has CPU cores, or more in some cases. Network bandwidth utilized per system will be the sum of the bandwidth utilization of all jobs on the system. In determining the appropriate network connectivity for your design, consider the number and type of jobs typical for each server within your environment.

End-to-End Latency

End-to-end latency refers to the round-trip Internet Control Message Protocol (ICMP) ping time between the two end points of a job. For VM migrations, the end points are two different physical compute servers. For all other job types, one end point is a compute server and the other end point is a storage server.

Ping latency varies with packet size. For this analysis, we use 64, 1500, and 9000 byte packet sizes. This corresponds to the majority of small packet sizes seen in production, traditional maximum Ethernet frame size, and **jumbo frames** maximum Ethernet frame size.

Each link in the chain from one end to the other adds latency. We will discuss the following components:

- Client NIC latency
- Switch latency
- Server NIC latency

Keep in mind that the switch latency is the total of all switch hops.

Both compute and storage servers use Network Interface Cards (NICs) to connect to the network. 10-gigabit Ethernet NICs have significantly lower latency than 1-gigabit Ethernet. This lower latency affects all job types.

Using Centos 5.6 and Broadcom BCM57711/BCM5721 NICs with copper cabling client NIC latency was shown to vary with packet size according to the following table.

Table 3: Ethernet Client NIC Latency

	64 Bytes	1500 Bytes	9000 Bytes
1Gb/s CAT-5	26 µs	60 µs	218 µs
10Gb/s Twinax	24 µs	29 µs	61 µs
10Gb/s SR Fiber	24 µs	29 µs	61 µs

Each switch between the end points (switch hops) will also add latency. This latency will vary from one product to another. Switch equipment marketing materials can be used to estimate this latency, but keep in mind packet size. In this analysis, the following table of top-of-rack and chassis switch latencies was compiled from a sampling of enterprise-class products available in 2011:

Table 4: Ethernet Switch Latency

	64 Bytes	1500 Bytes	9000 Bytes
Top-of-Rack Switch	1 - 4 µs	2 - 6 µs	3 - 4 µs
Chassis Switch	6 – 35 µs	8 – 100 µs	26 – 140 µs

Oversubscription and Bandwidth Contention

Bandwidth contention is the state of a specific network link where all available bandwidth is consumed by the traffic from two or more devices. This state will cause buffering of packets, resulting in latency variation (jitter), or in the worst cases, packet drops. Bandwidth contention increases in probability with higher oversubscription ratios.

Network oversubscription is the use of one network link to support the traffic from multiple network links with a greater total bandwidth capacity. This is generally done for the purpose of connecting one switch to another, without utilizing fully half of the switch capacity for uplink.

For instance, a top-of-rack switch with forty-eight 1-gigabit downstream ports and four 40-gigabit uplink ports has a minimum oversubscription ratio of 3:1. Oversubscription can be increased by decreasing the number of uplink connections. If only two of the 40-gigabit uplink ports are used, the oversubscription ratio becomes 6:1.

Oversubscription can be calculated on individual components, as in the top-of-rack switch example above or by looking at multiple network components. For instance, consider a network of 10 storage servers, each dual-connected at 10 Gb/sec directly to line-rate spine switch ports, and 48 compute servers each single connected at 10 Gb/sec to a top-of-rack switch, which is connected to each of the two spine switches at 80 Gb/sec. In this design, the compute servers are at an oversubscription ratio of 3:1. The storage servers are connected at line-rate (1:1 oversubscription radio). The overall oversubscription of the compute servers to the storage servers is 2.4:1.

Ideally, all network connections would be line-rate. In practice, oversubscription is a good way to balance cost with performance.

If storage server connectivity is designed as described in "Network Design Permutations" on page 6, oversubscription of storage server network ports is counterproductive. These network connections should be designed without oversubscription. This is known as **line-rate**.

Network Latency and Job Performance

All network traffic is negatively impacted by higher latency. The degree to which a grid job is impacted by network latency is approximately proportional to the amount of time the job spends performing network actions.

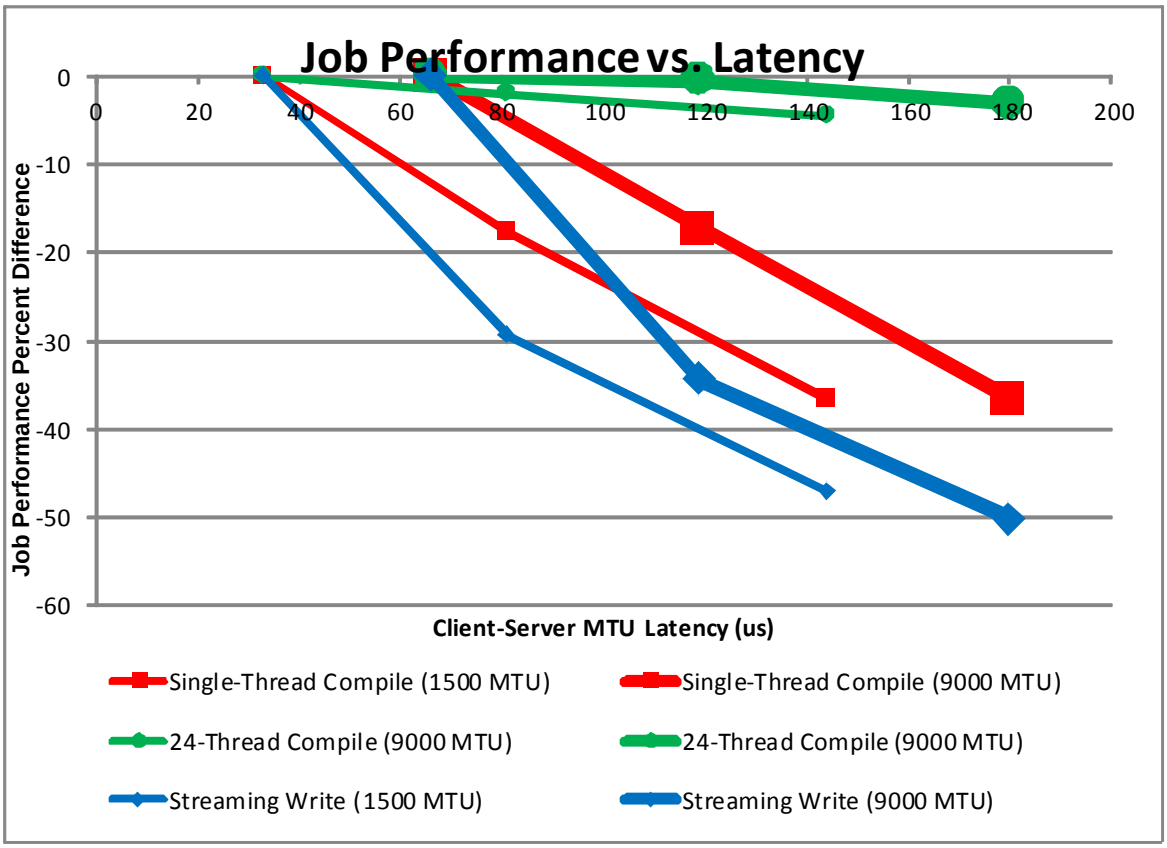

Figure 3. Job Performance vs. Network Latency

Figure 3 shows that higher latency negatively affects all jobs, but the single-thread software compile and streaming write are affected much more than the multi-threaded compile. The single-thread compile and streaming write wait on every NAS operation to finish before they can progress to the next step. The multi-threaded job can do local computation on some threads, while other threads wait for NAS operations to complete, reducing the performance degradation of higher network latency.

Generally speaking, the more I/O a job does, the more it is affected by higher latency, but asynchronous and parallel I/O can compensate to a large degree.

Maximum Transfer Unit Size

Ethernet Maximum Transfer Unit (MTU) size is the maximum size of the frame (Ethernet packet) on the wire. Increasing the MTU of a device allows it to put more data in each packet, lowering the ratio of payload to header data being transmitted, increasing efficiency and available bandwidth. Increasing the MTU setting of a device comes with some support difficulties:

- All devices within the same layer-2 domain (LAN, VLAN) must have the same MTU setting.
- Mixing very large packets with very small packets may cause increases in small packet latency, as they wait for the larger packets to be transmitted.

In practice, on today's Ethernet networks there are two choices for MTU: 1500 and 9000. Most Ethernet devices come from the factory with a default MTU of 1500. Some devices support 9000, which are known as jumbo frames.

The decision of whether or not to use jumbo frames on a production network must take into consideration the following factors:

- Do all devices on the target network support jumbo frames?
- Does the performance benefit outweigh the above support difficulties enabling jumbo frames?

To address the second question above, consider Figure 4. The use of jumbo frames makes a significant difference in job performance for I/O intensive jobs.

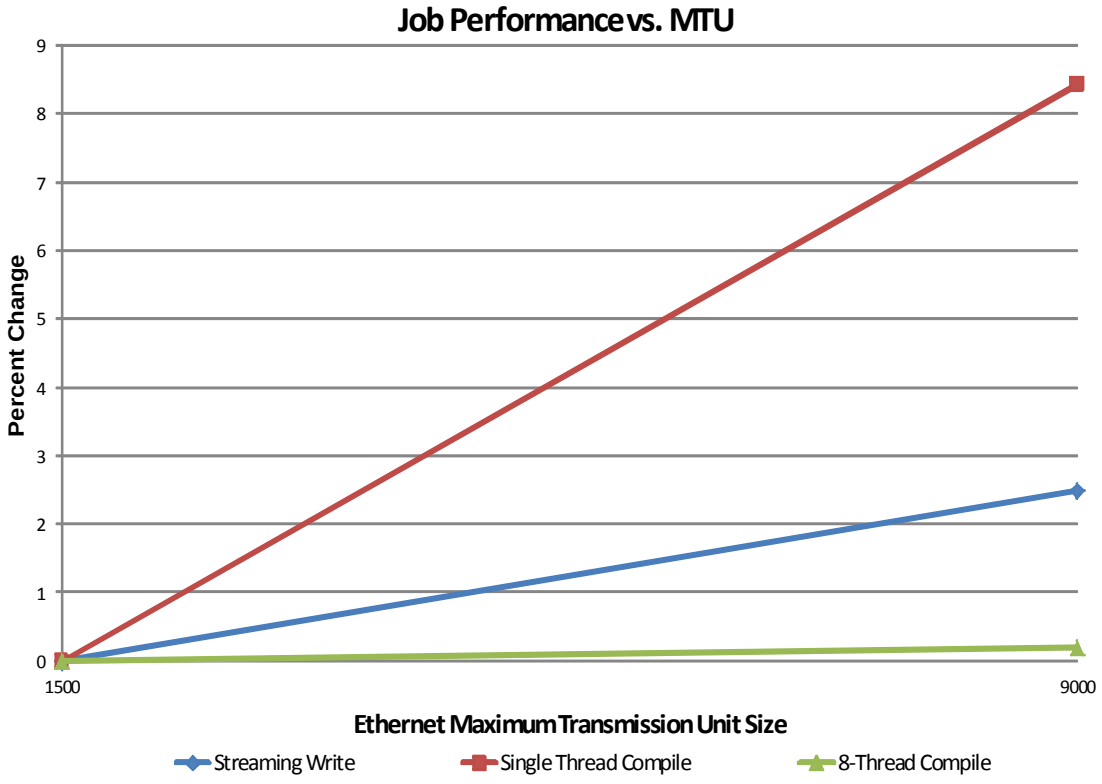

Figure 4. Job Performance vs. MTU

Conclusion

Network design decisions have a direct impact on job performance. We have observed that the following design elements have a positive effect:

- 10-gigabit Ethernet connected compute servers have increased bandwidth and lower latency compared to 1-gigabit Ethernet connected servers. This can increase job performance by as much as 130% for large NFS writes and as much as 190% for virtual machine migrations.
- Using jumbo frames (9000-byte MTU Ethernet networks) rather than the standard 1500-byte Ethernet MTU can improve large NFS write performance by as much as 2.5% and single-thread compile jobs by as much as 8%.
- A network designed for a minimum latency (40 µs in these tests), as compared to a typical 1-gigabit network of five years ago (150 µs), can increase job performance by as much as 50%.

Incorporating all of the above design elements results in job performance improvement ranging from 1%, for the least network-sensitive jobs (multi-thread compiles in this test), to 450% for the most network-sensitive jobs, such as VM migration and streaming writes.

In addition to improving performance, the above design elements can increase hardware and support costs. However, improving the overall performance of a grid computing system will reduce costs associated with job run time: customer time, power consumption, and tool license cost. These cost reductions can result in dramatic net savings.

Acknowledgements

Brendan Jacques assisted in all phases of this analysis, providing network expertise, hardware configuration and vendor management. This analysis could not have been completed without his great efforts.