Scale up: Building a State-of-the Art Enterprise Supercomputer

Sponsored By: HPC Cluster Solutions

Participants

Raytheon	AMD	Mellanox TECHNOLOGIES	DataDirect NETWORKS

About Appro International

Appro International, headquartered in Milpitas, CA, is a leading developer of high-performance, density-managed servers, storage subsystems, and high-end workstations for the high-performance and enterprise computing markets. Appro enables a variety of network computing applications by developing powerful, scalable, and reliable clusters, servers, and storage subsystems. We are committed to product innovation, quality, and customer service to build lasting relationship with customers.

Contents

Executive Summary

This reference document is based on a seminar on High Performance Computing (HPC) sponsored by Appro International. Other participant companies included Raytheon, AMD, Mellanox, and DataDirect.

Commodity clusters have become the primary choice for High Performance Computing (HPC) applications. Vendors are beginning to form alliances that allow them to offer complete turnkey solutions to users. It all starts with the design of the system; it must be viewed as a single entity to offer a balanced, scalable, and reliable platform to end-users. Its components must be well selected and well integrated. Appro, leading provider of High-Performance servers and clusters, brings together products from best-of-breed partners into an integrated solution that customers can use out of the box. Appro's partners work with Appro to optimize their products with Appro's products to ensure compatibility, high performance, configuration flexibility, and reliability and to deliver the best Enterprise and High Performance Computing integrated solution for this market.

Introduction

Appro offers Xtreme Blade solution and brings together the following partners and products:

- AMD's high-performance multi-core processors
- Mellanox's InfiniBand Fabric and I/O Interconnect products
- DataDirect's InfiniBand-based secondary storage products
- Appro's ClusterStack Pro (Raytheon's Software reference design)

This document describes these products in detail and then provides examples of supercomputer class systems built with these products. Appro's goal is to supply ready-to-use systems that are balanced and scalable over a wide range, and highly reliable.

Consider the Total Cost of Ownership

When considering a complete system, it is important to examine the overall computer architecture from the perspective of balance, scalability, and reliability. These factors are just as important as having fast CPUs. Power consumption and cooling are very important issues because substations and air handlers tend to be limited resources. Even small items such as cable routing are important. A simple approach is always best. A good rule of thumb is anything that is not required to get the job done is a maintenance problem.

Fault tolerance and on-line spares are the best approach to maintenance. Do not be tempted to trade low hardware costs for higher maintenance and support cost. An organization should put the maintenance schedule on that organization's terms. Mean Time To Repair (MTTR) should be less important than Mean Time Between Interruptions (MTTI) or Mean Time Between Failures (MTBF).

Redundancy is a key factor for large systems. When systems reach a certain size, it is not possible to gain the necessary availability without redundancy. Redundancy can be used to increase both reliability and system performance. For example, a dual-rail network fabric can double the performance while providing the redundancy necessary to ensure a system's availability. Multi-rail networks based on one technology are better than untested or unworkable plans to back up one type of network with another.

What is High Performance Computing?

During the last decade or so, high performance computing has been characterized as computing systems for high-end applications such as modeling, simulation, and analysis of complex physical phenomena. High performance computing tends to be CPU-intensive, requires a large amount of storage, and currently

requires performance in the teraflop region. In the past, these requirements have been met by using custom supercomputers. Today they are more often met by using clusters of commodity processors.

Strategies to achieve the high performance required by these applications depend on whether the primary need is for high capacity or high capability computing.

- "High **capacity** computing" refers to systems that run large numbers of low CPU count codes and strive for maximum throughput.
- "High **capability** computing" refers to systems that run a small number of high CPU count codes with the goal being minimum runtime and/or maximum scale-up.

High performance computing is currently used to describe nearly every high capacity or high capability system. Both areas are being taken over by commodity clusters.

The Problem with Commodity Clusters

Current commodity clusters have the following problems:

- Integration cost is too high
- Integration time is excessive
- MTBF is too low for large node count systems
- Ad Hoc approach to architecture and engineering
- Thermal, interconnect and power problems
- Most large HPC systems are one of a kind
- No total system view of designs

Integration costs have been too high as the result of vendors shipping a box of parts and expecting the customer to integrate those parts. The MTBF is too low because there is no total system view of the architecture. These problems have prevented second tier suppliers from taking more business away from first tier suppliers even though they have a significant price advantage.

View the Entire System as an Entity

A machine must be viewed as a whole and not as separate parts or subsystems.

The following list represents a small percentage of what must be considered in order to make the right decisions when building the architecture of a computer system:

- **System requirements:** performance (capacity and capability), power limitations, cooling limitations, floor space limitations, cabling limitations, availability, and maintenance
- **Interconnect fabric:** topology, technology, latency, bandwidth, scalability, redundancy, and packaging
- **Software requirements:** Reliability, Availability, Serviceability (RAS), resource management, operating system, scheduling, and diagnostics
- **Processor selection:** architecture, bandwidth, memory latency, memory balance, packaging, and cooling requirements
- **Memory:** architecture, bandwidth, latency, and error correction requirements

A computer system must be balanced and scalable in order for a large number of processes to work together or even for one process to work well. A system that has been designed with a total view of the requirements and technology is essential. No one can take software and hardware components and put them together for the first time and expect to have a complete plug-and-play solution without some difficulty. For most users a cluster is a tool, not a computer science project. Integrating a system requires

an architecture with well defined requirements. A system's components must be well selected to meet the requirements. The components must be tested. In this business, things do not always work as expected.

The components must be well integrated, which includes integrating the systems software components so they work as a single integrated package. Integration support includes:

- Reliability
- Availability
- Serviceability
- Resource management and scheduling
- Interconnect fabric routing and recovery
- Redundancy and fault tolerance

Availability

The maintenance goal for current generation systems should be to have the repair person on call, not onsite. It is too expensive to have onsite maintenance, even for large systems. This requires availability values of 97% or better. Achieving 97% availability with a two-hour Mean Time To Repair (which probably means a repair person is onsite) requires a Mean Time To Interruption value for a system greater than 61 hours. Achieving 98% availability with an offsite repair person and a 24-hour Mean Time to Repair would drive the Mean Time To Interruption requirement up to 1176 hours. This requires the Mean Time To Interruption value for a system to be on the order of 1000 hours or for interruptions to be handled by the system's Reliability Availability and Serviceability software. The RAS software can automatically restart the system using online spares. In this case, the onsite repair is done by software and the spares can be replaced later.

What Makes a Computer Scalable?

Making a computer scalable requires a balance in the hardware. Memory, fabric, and I/O bandwidth must match the CPU bandwidth. It is not possible to have a fast CPU with a slow fabric or a slow memory and have a fast computer. The architecture—including the interconnect fabric and fabric management—must be scalable. The software, as well, must be scalable. This includes RAS software, the operating system, libraries, tools, and applications.

Balance Requirements

Figure 1 illustrates the source of systems balance requirements.

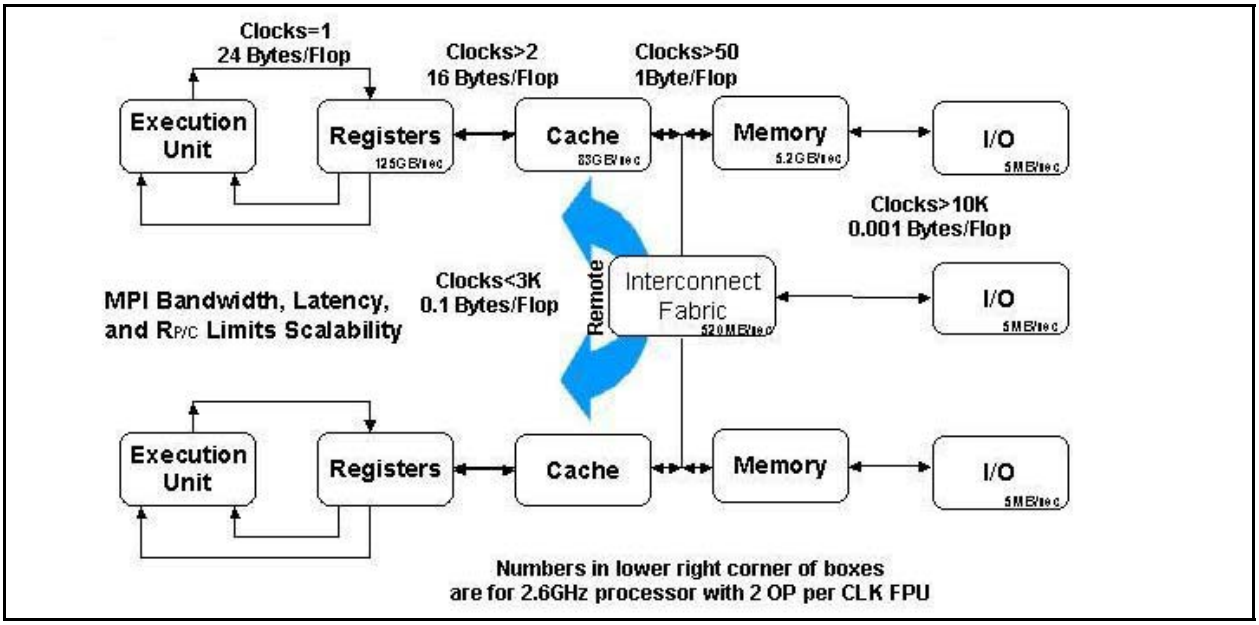

Figure 1. Where do balance requirements come from?

To produce one floating-point operation (FLOP), the computer's execution unit gets two 8-byte input values and produces an 8-byte result. That means 24 bytes have to move between the registers and the execution unit for one floating-point operation to be executed.

For a 2.6GHz processor that executes two operations per clock, the local registers must provide 125 gigabytes per second of bandwidth. At the L1 cache there is a reduction to about 16 bytes per floating point operation, which requires 83 gigabytes per second of bandwidth. Because the registers and L1 cache bandwidth are on-chip, the implementation is less difficult. The requirements for the off-chip memory still require one byte per FLOP or approximately five gigabytes per second to support a CPU running at this rate. The main memory bandwidth requirement is problem dependent and depends on the cache-hit ratio. To reduce the main memory bandwidth requirement by a factor of 16 requires an effective cache hit ratio of approximately 94%. This means that better than 94% of the time the data required by the execution unit is located in the local on-chip registers or cache.

When two or more computers run in parallel, the interconnect fabric presents a problem similar to a caching problem. Most memory references are to the local processor's memory, but the smaller percentage of references to remote processors' memory are delayed by the latency of the fabric that interconnects the processors and by the bandwidth of the interconnect fabric. The fabric in a well-designed cluster should have at least one-tenth of the main memory's bandwidth to meet the requirements of a capability computer. Five to ten times the minimum value is preferable. It is more important that the fabric have reasonable latency. It is preferable to have a latency on the order of 1,000 clocks or less, but greater than 3,000 clocks is more typical for current technology. Generally, what limits the scalability of a cluster is the ratio of the interconnect fabric speed and latency to the memory speed and latency.

Ideal Requirements for a *Capacity* Platform

For a capacity platform, the ideal requirements include very high single processor performance and balanced memory bandwidth. The memory bandwidth should be greater than one gigabyte/second per gigaflop/second with a latency of less than 50 CPU clocks. This level of memory performance was true but was reduced by the introduction of dual core chips with no increase in memory bandwidth.

In capacity machines, two-way, four-way or eight-way SMP nodes are desirable. The goal is to have a large memory capacity per node that is shared by multiple processors because many jobs may be running on each node. There is also a need for global, high-bandwidth storage systems in which any processor can be scheduled on any job and have global access to the storage.

For a capacity computer, the memory size should be greater than two gigabytes/second per gigaflop/second. The I/O bandwidth should be greater than one megabyte/second per gigaflop/second and secondary storage capacity needs to exceed 20 gigabytes/second per gigaflop/second.

Ideal Requirements for a *Capability* Platform

For a capability platform, the requirements change somewhat. A high-performance processor is still important, as is a balanced memory bandwidth. However, the interconnect fabric is much more important because the processors have to operate in parallel, which means placing greater emphasis on the latency and bandwidth between processors. Consider the following capability platform requirements:

- Memory bandwidth should be greater than one gigabyte/second per gigaflop/second with a latency of less than 50 CPU clocks for the memory.
- Interconnect fabric latency should be less than 3,000 CPU clocks. The interconnect fabric bandwidth should be greater than 0.1 gigabyte/second per gigaflop/second with five to ten times the minimum bandwidth being desirable.
- High-performance message passing interface (MPI) or global memory support is needed in order to run parallel jobs that will scale. An MPI package should process status and support alternative paths.
- Large memory capacity per system is required for large jobs. The memory capacity should be greater than 0.5 gigabyte/second per gigaflop/second. This is generally smaller than for a capacity system since processors and memories are dedicated to single jobs.
- The I/O bandwidth should be greater than one megabyte/second per gigaflop/second. High bandwidth in a capability system is used for check pointing and for moving large data sets. Check pointing is a critical issue because many capability jobs are long running.
- Global high bandwidth secondary storage is needed. Storage needs to be global because all the processors need access to the storage in parallel. The secondary storage should be capable of sustaining a large number of parallel operations. Secondary storage size should be greater than 10 gigabytes per gigaflop/second.

Why the Memory System Should Be Balanced

This document has discussed memory bandwidth in terms of gigabytes per gigaflop/second. The main memory bandwidth that is required depends on the CPU performance and on the cache hit ratio sustained by typical programs running on the CPU.

Figure 2 shows the performance vs. effective cache hit ratio for memories with a ratio of 0.25 gigabyte/second per gigaflop/second up to 1.00 gigabyte/second per gigaflop/second for a typical microprocessor. A slower memory causes performance to drop more rapidly as the cache hit ratio decreases.

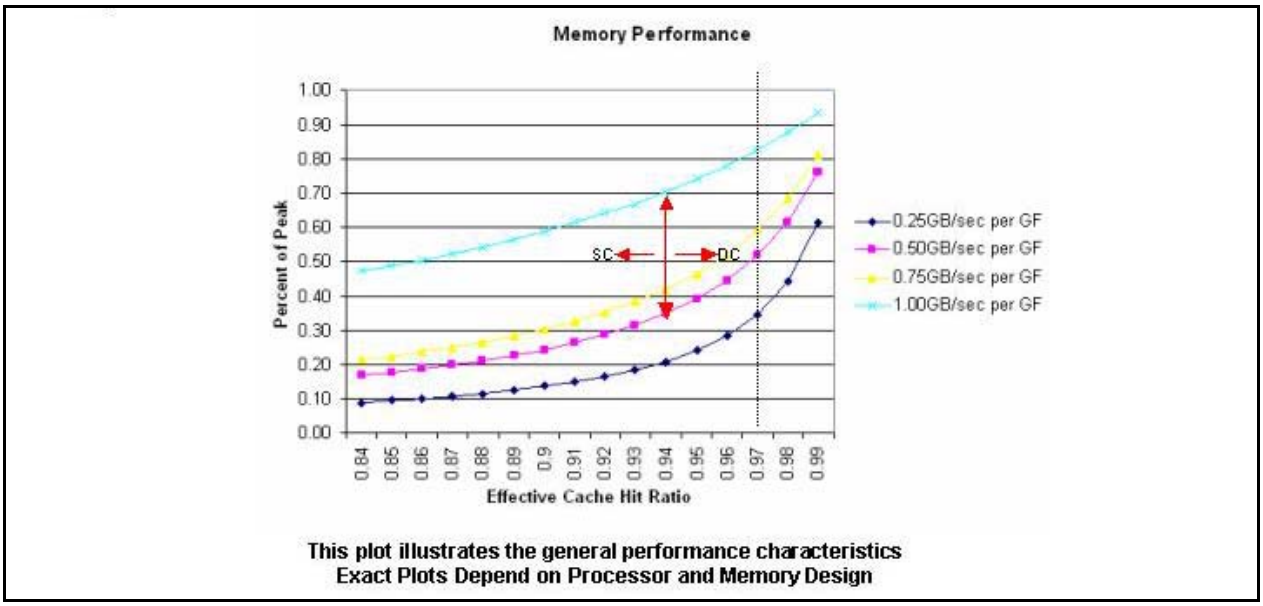

Figure 2. Balanced memory performance

Most current generation dual-core processors have between 0.5 and 0.6 gigabytes/second per gigaflop/second of memory bandwidth. Most problems fall to the left of the vertical dotted line in Figure 2 because achieving cache hit ratios higher than the high 97% is difficult for most computer codes. Consider what happens as an HPC program places greater demands on the main memory because the hit ratio is low. For the 0.5 gigabyte/second per gigaflop/second curve shown in Figure 2, the vertical red line indicates a point at which it is a better strategy to shut off one of the cores in a dual-core processor and run only one core in the chip. This is true because there is a two-to-one difference in the two curves indicated by the vertical red line. Programs to the left of the vertical red line will run better using the full memory bandwidth for the one core.

Transaction processing and file server programs running on dual-core AMD™ and Intel® processors might be to the right of the vertical red line shown in Figure 2. Unfortunately, most HPC problems may be to the left of that line. It is important to have a scheduler that can be directed to run only one core per chip for specific problems.

The Challenges of Scaling Runtime

Amdahl's Law, illustrated in Figure 3, shows that runtime is difficult to scale for codes with significant serial content. Serial code is code that can be executed on only one processor.

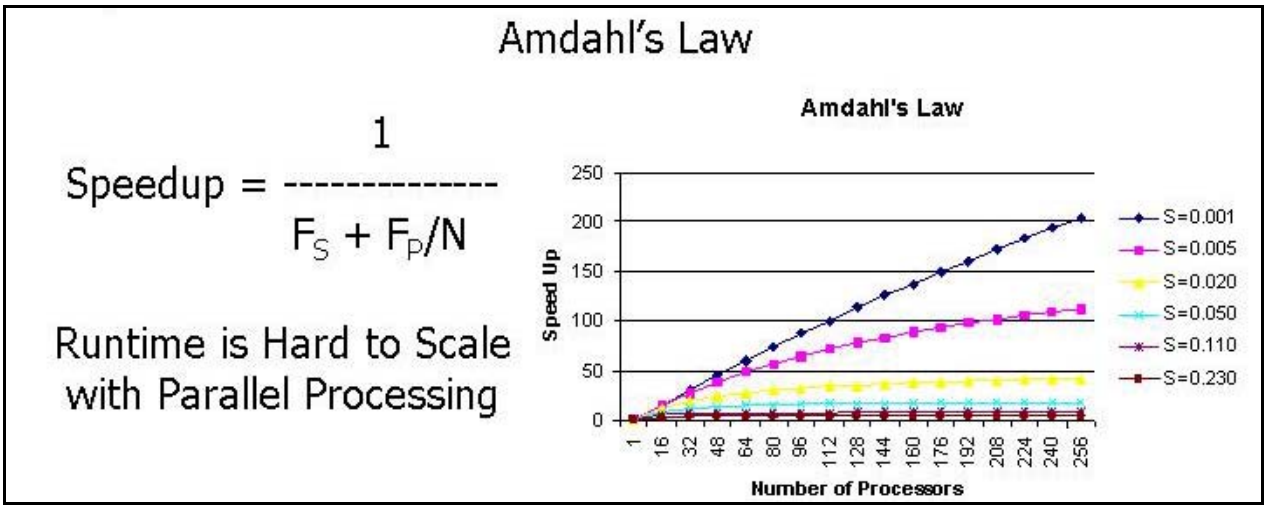

Figure 3. Scaling—runtime

When there is a lot of serial code that can only execute on a single processor, a job will not execute much faster with parallel processors. The only improvement in execution time will be a reduction in the time required to execute the part of the code that can be subdivided to run in parallel. The time required to run the serial code will remain the same. Using the smallest number of the fastest processors available is the best solution in all cases and especially for codes with significant serial content.

However, the scope and resolution of most problems scale well with parallel processing, as shown in Figure 4.

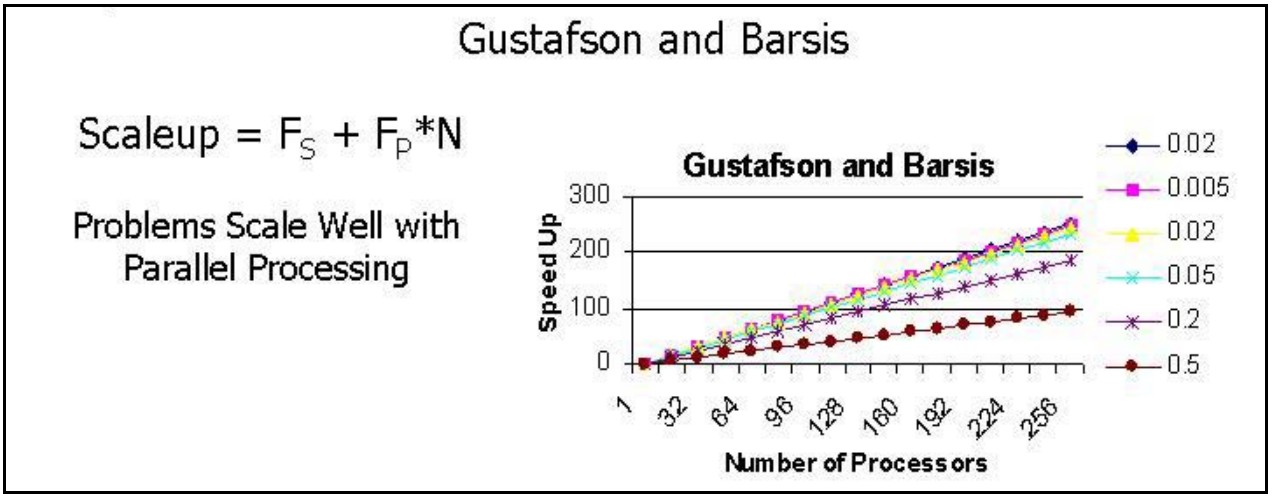

Figure 4. Scaling—problem size/resolution

Problems scale well because it is possible to increase the resolution or the size of the problem and keep the execution time nearly constant. The same amount of time may be required, but it is possible to process a lot more data. In some cases it may be possible to reduce the runtime while increasing the amount of data being processed.

Runtime can be decreased, but the increase in speed in terms of throughput will not be as dramatic as the increase in scalability or volume of data processed. Parallel processing fits the way problems scale.

There is a slowdown associated with the interconnect fabric between processors. The ratio of the remote-to-local memory access is the culprit that slows most problems. Therefore, fabric performance is critical to scaling massively parallel systems. This is why a well-balanced fabric is necessary.

The equation that explains this decrease in performance is:

Slowdown = $(F_L + F_R R_{R/L})$

where:

F_L = Fraction of local memory references

F_R = Fraction of non-local memory references

$R_{R/L}$ = Ration of remote-to-local-memory access time

Balance and Scalability

For most parallel applications, performance is determined by parallel scalability and not the speed of the individual CPUs. A well-balanced architecture is less sensitive to communications overhead. A system with weak communications can easily lose most of its performance for communications-intensive applications. There must be a balance between processor, memory, interconnect and I/O performance to achieve overall performance. Weak memory and interconnect performance limit the capability of most existing commodity clusters. It is normal for large parallel applications running on large clusters to achieve less than 10% of the peak performance that would be predicted by multiplying the single processor performance by the number of processors. The slowdown for most problems is a result of low fabric performance. Best performance is still achieved by using the smallest number of the fastest processors available. This is why there has been such an intense interest in higher clock rates. To offset a 2X difference in clock rate it may typically require four times as many processors because of non-linear scaling.

Processor Design

Differences in processor design must also be taken into account. Processor designs such as the Intel Pentium4 are designed for maximum clock rate operation. This makes the design have fewer logic decision levels between clock steps. In other words, complex instructions take more clocks to execute. In the Pentium design, simple fixed-point operations are very fast compared to other processors, like the Itanium, but more complex operations may take up to two times as many clocks to execute. An AMD Opteron™ processor running at two-thirds the clock rate matches the performance of an Intel® Pentium4. An Intel Itanium can match the Pentium running at approximately half the clock rate. Another difference in processor designs is the configuration of floating-point units. The Intel® Pentium® and AMD Opteron™ have floating point arithmetic units that can execute two operations of the same type in parallel if the operands can be assembled as a vector. This can be an advantage when executing the same sequence of operations on strings of operands. Some microprocessors have separate floating point add and multiply units. The two units can run in parallel if independent operations can be scheduled. Other microprocessors have one or two so-called fused floating-point units that can perform a multiply, add, or a chained multiply add in a single operation. Microprocessors like the IBM P4 and P5 have dual fused multiply add units. These processors claim four operations per clock step. The only problem is that codes seldom consist of all dot products. These machines generally get no better than approximately 2.2 operations per clock step, which is good but nowhere near the four they get on specific benchmarks. Some benchmarks are better advertising than they are indicators of performance. The best performance per dollar still favors the Intel Pentium and AMD Opteron™. While the Intel Itanium and IBM P Series outperform the Pentium® and Opteron™, the cost is much higher than the mostly small difference in performance warrant.

Reliability

The individual hardware components in commodity clusters are very reliable when considered on a one-by-one basis. For example, a motherboard used in a desktop may have an MTBF of 10,000 hours. However, if the same motherboard is used in a 1000-processor cluster, one is likely to fail every 10 hours on average.

Disk drives range in reliability from 250,000 to 1.2 million hours. One factor that makes disk drives less reliable is putting a disk drive in a hot environment. When a disk drive is put inside a 1U chassis with 200-watt-plus CPUs, it will fail much more frequently than the advertised MTBF value.

Power supplies have an MTBF of approximately 40,000 hours, and this usually does not include fans, which have an MTBF of 20,000 hours or less. The real MTBF for power supplies is more on the order of 10,000 to 15,000 hours, depending on what kind bearings are used in the fans. Fans with bushings that are designed for desktop applications fail much more frequently than fans with ball bearings. Only fans with ball bearings should be used in server applications.

Systems with thousands of processors require so many components that the aggregate Mean Time To Interruption may be unacceptable. Therefore, redundancy and the ability to recover quickly from failures are essential. Redundancy and recovery must be built into the system architecture to ensure acceptable availability.

Improving Reliability

Take the following precautions to improve the reliability of components:

- Do not put extra components in a computer. Extra components on a server board may diminish the reliability more than any imagined benefit provided. This is especially true with so-called management features.
- Make power supplies and fans for a cluster of any scale redundant. These are the most frequent components to fail. For a system with 576 non-redundant power supplies, the MTBF would be on the order of 14 hours. With 3+1 redundancy, the MTBF should be greater than 37,000 hours and the MTTI would be millions of hours if failed units are replaced on a timely basis.
- Do not locate disk drives in the same enclosure with high performance CPUs. Disk drives will fail much more frequently at higher temperatures. Increasing the operating temperature of the disk drives dramatically decreases the MTBF.
- Consider whether checkpoint recovery is necessary. Checkpoint recovery is not practical in 1U nodes with disk drives in the node because it is not possible to store data on the node that fails and still recover that data. RAID (redundant array of independent disks) groups are impractical in 1U nodes.
- Require that the network fabric be redundant and fault-tolerant to support non-stop operation. This is the one part of a cluster that must be reliable. The network fabric in a system may have the greatest overall impact on both system availability and performance. It is possible to replace a failed processor and checkpoint-restart a failed computer, but it is not possible to checkpoint-restart a failed network fabric. Confirm that the file system supports fault tolerance and recovery. A file system should be designed to eliminate or minimize interruptions and above all ensure that there is no loss of data. The availability for the secondary storage should be 99.9% with a guarantee of no data loss. This requires two copies of all data. Tape archives have traditionally performed this function but low cost disk storage is now being used for archive (second copy) storage.
- Configure the servers in RAID parity groups or make them redundant to ensure uninterrupted operation.

- Configure disk drives into RAID arrays to prevent losing data because of disk failures. This ensures continuous data availability. This does not protect against software that corrupts the data. Only a second archive copy provides this protection.
- Have cluster management software to run the entire system. The cluster management software should manage all aspects of cluster operation. Do not depend on several independent software programs that are not integrated. The cluster manager should work with the systems scheduler to support checkpoint-restart, as well as recovery from processor failures.
- Make the management and communications networks redundant.
- Verify that the RAS software can replace failed components with spares when available.

Building a High Performance Computer System with Commodity Parts

Once a system's performance issues are thoroughly understood, commodity parts can be used to build high performance computing systems that are balanced, scalable, and reliable. This section describes the design goals that should be consider when building such a system.

The first goal is to build the best system for the lowest cost. This means building a standards-based system using all commodity hardware and software components. There should be no custom components or protocols. It is undesirable to use custom protocols even if they provide significant performance improvements because one-of-a-kind protocols tend to have short lives and are expensive to maintain.

The second goal is to achieve the highest possible performance. This can be accomplished by getting the highest possible single processor performance. If a system uses CPUs that have slower clock speeds because it is easier and more convenient to package those CPUs together, it is also necessary to consider the overhead of scaling. A computer with processors that are half as fast might require four times as many processors to get the same performance. It is important for that organization to remember that when building a machine with capability, everything must scale.

The third goal, which may be the most important, is to achieve the highest reliability possible. Reliability is essential when it is necessary to have a lot of CPUs. When purchasing commodity components, it is important to examine where to apply the redundancy. Plans for high reliability should include:

- Minimized complexity
- Minimized parts count
- Redundant fault-tolerant interconnect fabric
- Redundant power
- Redundancy in the secondary storage

The final goal is to achieve the highest functionality possible. To achieve this goal, use standards-based interfaces and protocols, commodity operating systems, commodity software, and an integrated cluster management software suite.

The Processor: Multi-Core Processor

Improving clock speed is not always a workable strategy to improve CPU performance. The issue is not that it is technically impossible to produce a product that is faster than about four gigahertz. Instead, from an economic standpoint, at some point the part becomes inefficient, for example, from a power and cooling perspective. This means it is no longer as cost effective to use a single-core CPU system.

A better solution is to reduce the clock speed and add one to three cores to a CPU. This approach results in improved performance at lower clock speeds.

The two biggest challenges that organizations face are:

- The power being delivered to data centers
- The cooling capacities in data centers

If no one pays attention to performance per watt, it is possible for an organization to have a great solution that cannot run because there is insufficient power and cooling. Once efficient systems are developed, it can be a good strategy to partition them appropriately.

Commodity CPUs are becoming dominant. The commodity CPU market is reducing the number of weeks or months required to perform configuration, installation, and tuning.

The trend in the industry, whether buying a notebook computer, a desktop or a server is the same. It is not possible to keep adding larger and larger amounts of cache to a die in order to improve performance based on limited architecture. The result is a die that is very big with a lot of transistors that draws more power and generates more heat.

AMD's Direct Connect Architecture allows developers to create robust solutions by connecting memory directly to the CPU. An on-chip memory controller reduces memory latency significantly, which allows Raytheon, Appro, and other partners to develop and build a highly robust solution.

As already mentioned, increasing clock speed is not a good solution. Instead, every time a core is added to the CPU, this provides the ability to slow down the processor. This results in drawing less power and generating less heat while still delivering incremental improvements in performance.

Standard two-way nodes now have four threads, and four-way nodes have eight threads and 128 gigabytes of RAM. Each AMD-based CPU core has its own exclusive set of cache memory.

AMD's dual-core parts fit in the same socket as a single-core part and draw the same amount of power. This is critical because that makes it possible to improve the capabilities in a given blade, such as the Appro Xtreme Blade™ Server, without changing the power requirements. This can improve performance or solve the same single-threaded problems that a single core will solve.

If any given problem is speed related, the solution is to use single cores. Organizations also can run dual cores for problems that require multiple CPUs. In taking this approach, organizations can use the same architecture on the same board. The same motherboard can support both single- and dual-core CPUs. This means a company like Appro can build a solution that is cost effective and provides value without having to monitor and maintain two different systems.

In the past, the solution was to use a memory controller hub where the memory controller resides. That is a central focal point for all I/O whether that I/O is visual, disk, network or memory. As a result, this creates conflicts and contentions on a single bus, which then results in higher memory latencies. This results in a bottleneck because traffic literally moves in either direction, and in many cases traffic has to stop and shut a gate to move the information in the right direction and then open the gate again.

AMD's HyperTransport™ technology interconnect is a point-to-point protocol that is universal. Links are either coherent or incoherent. A coherent link is a link that can communicate chip-to-chip without intervention from the CPU itself. A crossbar switch handles this link. In other words, a memory request from this device to the memory module occurs through that chip without much intervention at the CPU level. Memory has its own bus, which promotes good memory performance.

However, using multiple cores provides twice as much data traveling over the same bus. Using a four-way system would mean having two chips or physical sockets using a single bus. Even worse, it is necessary to slow down the bus in order to split it.

The point is that using a dual-core CPU results in having four logical CPUs using a slower bus to try to gain access to memory. This results in very poor memory performance. And yet, this worst-case scenario is still significantly better than using a single-core CPU. Memory bandwidth is critical.

AMD's PowerNow!™ Technology reduces CPU demand. This is achieved by enabling the CPU to optimize power by dynamically switching among multiple performance states (frequency and voltage combinations) based on CPU utilization without having to reset the CPU. AMD PowerNow! Technology lowers power consumption without compromising performance. It strengthens the AMD Opteron™ processor performance-per-watt capabilities. In other words, it delivers performance on demand, minimizing power when full CPU performance is not necessary.

For example, suppose an organization runs an operation about 10 hours a day on a 120-node system. Using AMD PowerNow! Technology would save about $25,000 each year in power consumption. It can provide up to 75% power savings.

Power is critical because it impacts both density and performance within a system. Along the same lines, the heat generated by parts is critical, and AMD processors generate a lower amount of heat.

The Storage Challenge: Native Attach InfiniBand

DataDirect Networks provides high performance, large capacity network storage solutions, including the S2A9500 product, which is a Fibre Channel 4 (FC-4) and InfiniBand modular RAID storage networking system.

The goal is that when single or multiple drive failures occur, Remote Direct Memory Access (RDMA) and low-latency file systems can still be accessed. One solution is the InfiniBand-enabled product for high-performance computing and clustering.

The S2A9500 product is a good solution when customers need multiple gigabytes per second. Scalability is an issue in terms of capacity and performance. DataDirect Networks can put over 1,000 drives behind a controller pair. That reduces an organization's costs because it is not necessary to buy multiple sets of controllers.

Components have power supplies and fans, which can cause problems. DataDirect Networks reduces that risk by putting a single drive in each drive tray for these tiers.

A feature called PowerLUNs provides the ability to strike across multiple tiers on the back end of this device. For example, InfiniBand runs at 860 megabytes per second on a single data rate, so that is a single port. It is impossible for disk drives to deliver that much in a single line. Organizations can get host software and perform striping, but host software striping usually costs money. PowerLUNs is provided free of charge.

DataDirect Networks also RAIDs the cache. Individual disk cards (Field Programmable Gate Arrays or FPGAs) control the back-end disk ports effectively or dual ported. There are five cards per controller that plug in the PCI-x, and a RAID is performed on the memory on each board in a 4+1 RAID configuration. This means it is impossible to lose data if a card fails or if a memory chip fails on a card.

Parity is calculated on the fly for all I/Os, reads and writes to provide quality of service and better data integrity. Benefits for shared file systems include:

- Native InfiniBand and RDMA (and FC-4)
- A great reduction of file system complexity and cost
- Huge performance reduces storage system count
- PowerLUNs reduce the number of LUNs and host striping required
- Support for more than 1000 fibre-channel disks and Serial ATA (SATA) disks

- Parallelism provides inherent load balancing, multi-pathing and zero-time failover
- Support for all open parallel and shared SAN file systems
- Support for all common operating systems

DataDirect Networks has active-active controllers. When running coherency mode, it is possible to access any back-end device, tier or LUN through any front-end host port with zero latency without requiring any failover software in the host. For example, it is easy to access a load balancing multi-pathing type driver. It is possible to send I/Os to LUN 0 across all eight front-end ports whether those ports are InfiniBand or fibre channel with zero latency failover.

Global clustered file systems leverage the client/server model. They take advantage of high-performance networking interconnect and messaging protocols, including InfiniBand, Myrinet, Quadrics, GIgE, files services, MPI, and others. Global clustered file systems use standard storage channels and protocols, which are mostly fibre channel and SCSI socket interfaces.

The standard way of handling I/O involves the I/O being broken up by the mid-SCSI layer in the kernel. Because two-megabyte I/O cannot go directly through the kernel to the RAID, this adds latency to the overall I/O, which reduces overall performance. The solution is to directly attach the I/O into the InfiniBand fabric and try to bypass the kernel and/or the mid-SCSI layer issues and take that two-megabyte I/O directly from the clients directly to the RAID.

With the 2.6 Linux kernel, SCSI generic drivers can be used to bypass problems and can result in a two- or four-megabyte I/O if passed from a client directly to the RAID. And that helps significantly with the performance. However, the goal is to attach directly into the fabric and go directly into the RAID cache and effectively to the RAID itself. This type of enabling requires data RDMA host interfaces, meaning, InfiniBand and possibly a 10-gigabyte Ethernet and a fibre channel.

The Networking Components: InfiniBand

Next, consider the fabric options. Ethernet was designed as a robust communications protocol. InfiniBand was created by industry leaders who foresaw the problem of unbalanced systems. InfiniBand is a hardware transport protocol designed for high-performance switched fabric for server and storage clustering.

One InfiniBand benefit for server clustering is the CPU and the operating system does not have to be involved in node-to-node application communication. InfiniBand can assume that responsibility. This allows the CPU to apply more cycles to the application, thus increasing overall performance.

InfiniBand also supports quality-of-service (QoS) features which will make it possible to partition and deliver guaranteed service levels. InfiniBand was designed to enable and accelerate grid or compute cluster environments because it provides an architecture that can deliver partitioning and quality of service on a packet basis.

InfiniBand allows users to create a scalable clustered environment where it is possible to scale to larger number of nodes and still get very high efficiency. It is a reliable, no-loss, self-healing fabric that includes automatic path migration. InfiniBand is well suited for customers who need affordable supercomputing.

Some of the biggest challenges organizations face when putting clusters together is the amount and complexity of wiring needed for a clustering solution. This happens when organizations have separate wires for different data types. This can include a storage wire (typically FC), a management wire (typically 100 BaseT), a communications wire (typically GbE), and a cluster wire (typically GbE).

One benefit of InfiniBand is that all four requirements can be delivered over a single InfiniBand wire. This is possible because applications use standard APIs, whether TCP, SCSI, or MPI. The InfiniBand community has developed InfiniBand equivalents. For example, SDP, IPoIB, SRP, iSER, NFS/RDMA, and Open MPI are available.

InfiniBand offers the highest bandwidth, the lowest latency, and low CPU utilization. It provides a balanced interconnect solution that can be used as part of a balanced supercomputer.

InfiniBand is a very efficient protocol and gets about 970 MB/sec using a 10Gb single data rate (SDR) HCA. Mellanox offers 10Gb single (SDR) and 20Gb double data rate (DDR) InfiniBand HCAs today. Mellanox plans to introduce 40Gb quad data rate (QDR) InfiniBand products in the future.

Today, InfiniBand customers use copper CX-4 cables. CX4 cable lengths up to 10m are supported for DDR. CX-4 cables are a less expensive solution compared to fiber optic cables. This can be an important consideration for large clusters. Industry is looking at optical InfiniBand solutions when longer distances are required

Network Topologies

The next step is to consider which network topology to use because different topologies offer different advantages, depending on the system. The topologies that can be used are:

- Mesh
- Torus
- Fat-tree

The topologies described in this section are recommended for different kinds of systems. A mesh topology, shown in Figure 5, is a good solution for large systems.

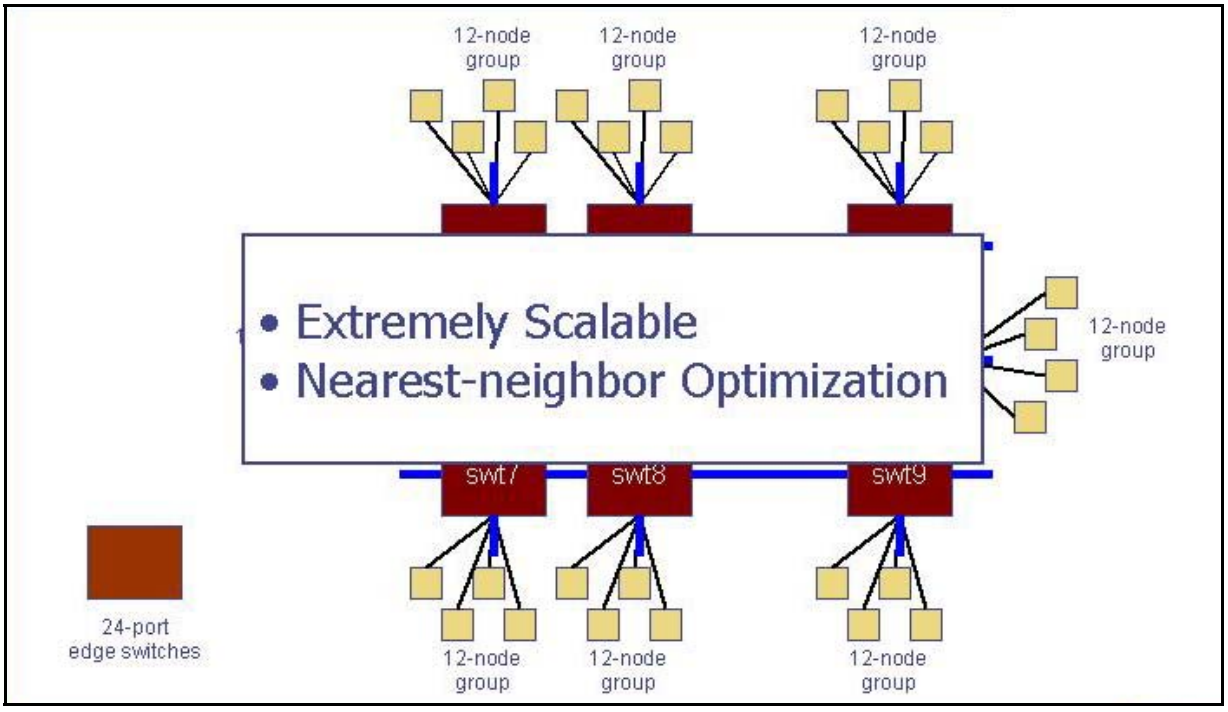

Figure 5. Mesh topology

A mesh topology matches parallel codes. It scales linearly. It has short point-to-point connections, and it is possible to use copper connections with high bandwidth. A mesh topology requires a more complex scheduler because nearest-neighbor problems and processes have to be located in the hardware as nearest neighbors.

A Torus topology, illustrated in Figure 6, is better for the highest bandwidth systems when the entire computer is used for a single code.

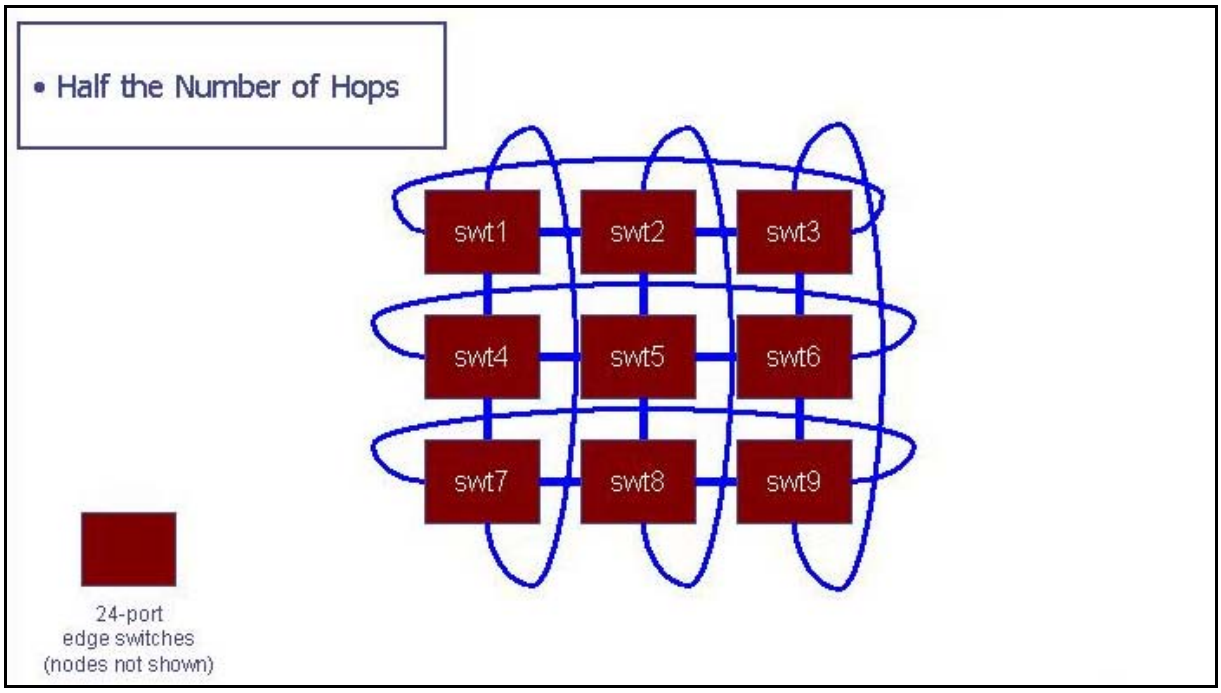

Figure 6. Torus topology

A Torus topology folds all the dimensions. If a much smaller code is located in the middle of the Torus, the only advantage might be the ability to wrap around the end to get enough processors in a single area or volume to execute a job. This topology does not present any advantages for small codes located in the middle of the array, but when a large code occupies the entire machine, a Torus topology is a good solution. It improves the connectivity and redundancy of a mesh topology with no added cost other than for more complicated routing software.

The Fat-Tree topology, shown in Figure 7, is the de facto standard.

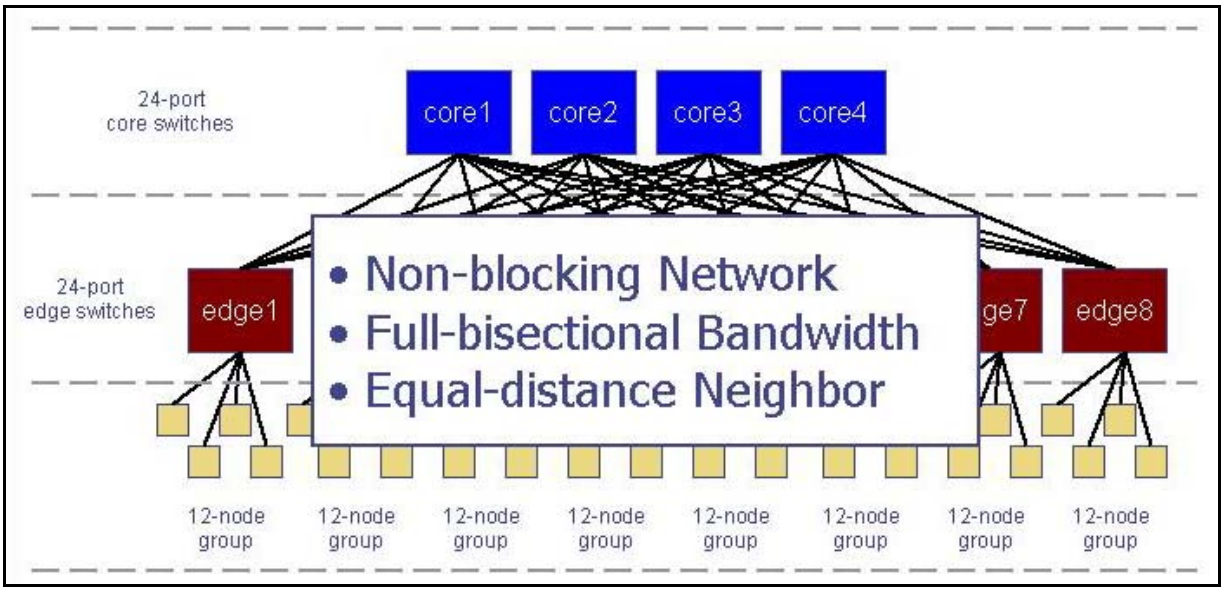

Figure 7. Fat-tree topology

A fat-tree topology is non-blocking, works well in small systems, is supported by open-source InfiniBand (Open IB), and does not require a smart scheduler. However, a fat-tree topology does not scale linearly. It also does not make efficient use of switches and cables. Furthermore, it requires fibre optic connections for large systems, especially at higher data rates. Because the length of the cables will be constrained at double and quad data rates, copper can only be used on short connections.

Using a Fat-Tree Switch

Figure 8 illustrates a simple Dual-Rail Fat-Tree switch.

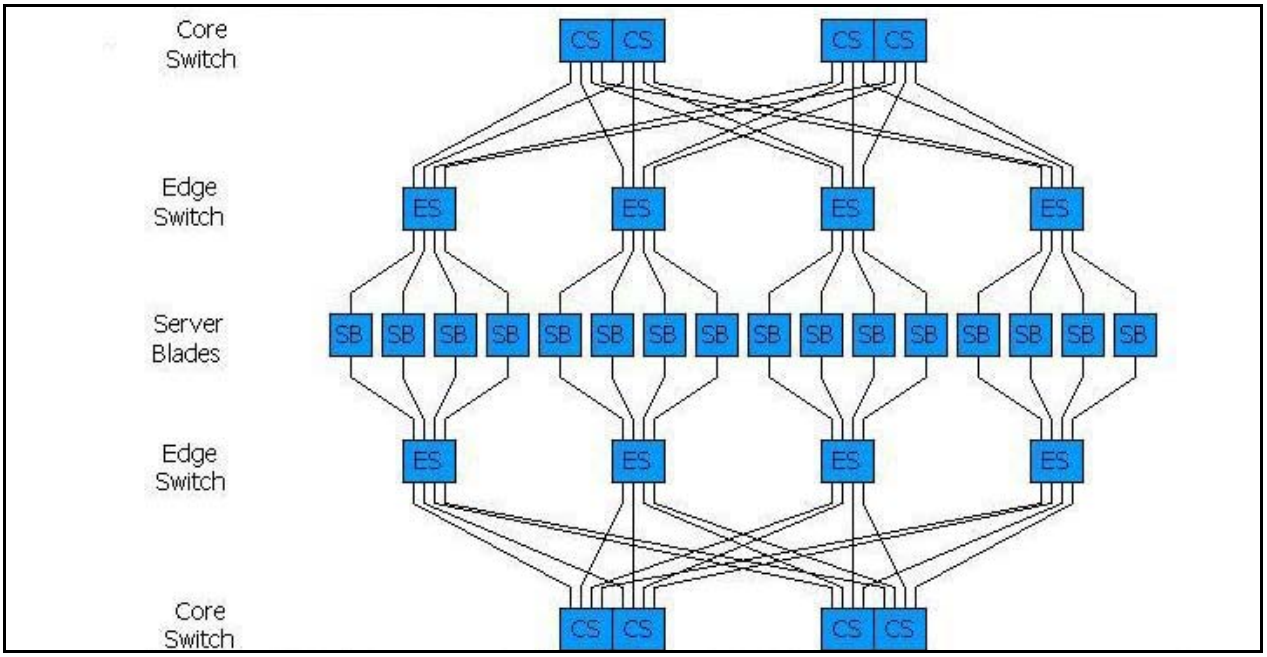

Figure 8. Fat-tree switch

Notice that the server blades (SB) and edge switches (ES) are built into the Appro Xtreme Blade Server. The core switch elements (CS) must be provided by external switches. A second fat-tree network can be used to provide twice the bandwidth and redundancy, as illustrated in Figure 8. This requires dual ports on the blades, and twice the number of fabric switches.

One fat-tree network in the example in Figure 8 costs a total of four edge switches and two core switches. A dual-rail fat tree would require eight edge switches and four core switches.

The Scheduler's View of the Nodes: Using a Mesh Topology

There are advantages and disadvantages to having four processors on a blade. An advantage for capacity problems is that the four processors can share the codes. The disadvantage is that they share the I/O bandwidth and memory bandwidth. For capability systems, the four processors can be viewed by the scheduler as a two-by-two array with a common amount of bandwidth to the outside.

In regard to nearest-neighbor codes, 30% to 40% of that traffic is consumed inside the node. This is important because it reduces the amount of external bandwidth required by the node. This effectively increases the amount of bandwidth across this interface. Therefore, the peak fabric bandwidth per processing element under these circumstances is about 0.25 gigabytes/second per gigaflop/second, which is better than most large clusters. This meets the requirements for a capability system for anything but the most communications-intense codes.

For parallel applications, the scheduler can also view the 12 blades within a sub-rack as a three-by-four array. Again, the nearest neighbors' communications can help to lower the effective bandwidth.

Mesh Interconnect: Routing Constraints

In a mesh topology it is possible to have routing deadlocks. One way to prevent deadlocks is by using dimension-based routing, a feature in the ClusterStack Pro software. Messages are routed to completion first on one dimension and then on the other. As a result, when routing out of a node, most of the traffic goes out on one axis. Because the InfiniBand protocol supports virtual circuits, it is possible to eliminate the balance problem by using one virtual circuit for messages that start on one axis and a different virtual circuit for messages that start on the other axis. When routing in a Torus, different virtual circuits are required for messages that route around the end of the array. This is required to prevent yet another kind of deadlock unique to the Torus topology. With these constraints, a mesh architecture still provides more bandwidth than needed for a capacity system and provides very acceptable bandwidth for a capability system, especially considering that most large parallel problems from the real world are nearest-neighbors problems.

The Server: The Appro Xtreme Blade

If it is not possible to have redundancy in the compute nodes (blades), it is possible to have redundancy in all other parts of the system.

The Appro Xtreme Blade Server is based on three technologies that provide a solid foundation for performance and reliability:

- AMD Opteron as the processors with low power options
- InfiniBand for the I/O with redundancy
- PCI Express as the interconnect technology for maximum performance
- Gigabit Ethernet with redundancy for management and external communications
- Redundant Power Supplies with redundant AC inputs
- Redundant cooling with high capacity ball bearing fans

InfiniBand provides high bandwidth and stability to support MPI applications, as well as storage high bandwidth secondary storage units such as those provided by DataDirect.

A full size rack can hold up to 72 blade servers. Each blade is a two-socket server. Up to 12 blades are mounted in a sub-rack that contains redundant InfiniBand switches, redundant GBE switches, redundant power supplies, and redundant cooling fans. There are up to six sub-racks mount in a rack. The redundant power supplies, fans, InfiniBand switches, and Ethernet switches are located in the back of the sub-rack. The idea is to produce a building block that is fully redundant and hot-swappable in order to provide high availability.

The Xtreme Blade offers a peak performance of 1.4 teraflops with 150 gigabytes/sec of sustained memory bandwidth. The required power is about 25 kilowatts maximum for a rack with 72 blades and 288 processors at 100% utilization. That number can be lowered, depending on a data center's requirements by using fewer blades or lower power Opteron™ processors. Using the new low-power version of the AMD processors lowers the power per rack by almost 50%.

Figure 9 shows an Xtreme Blade compute node.

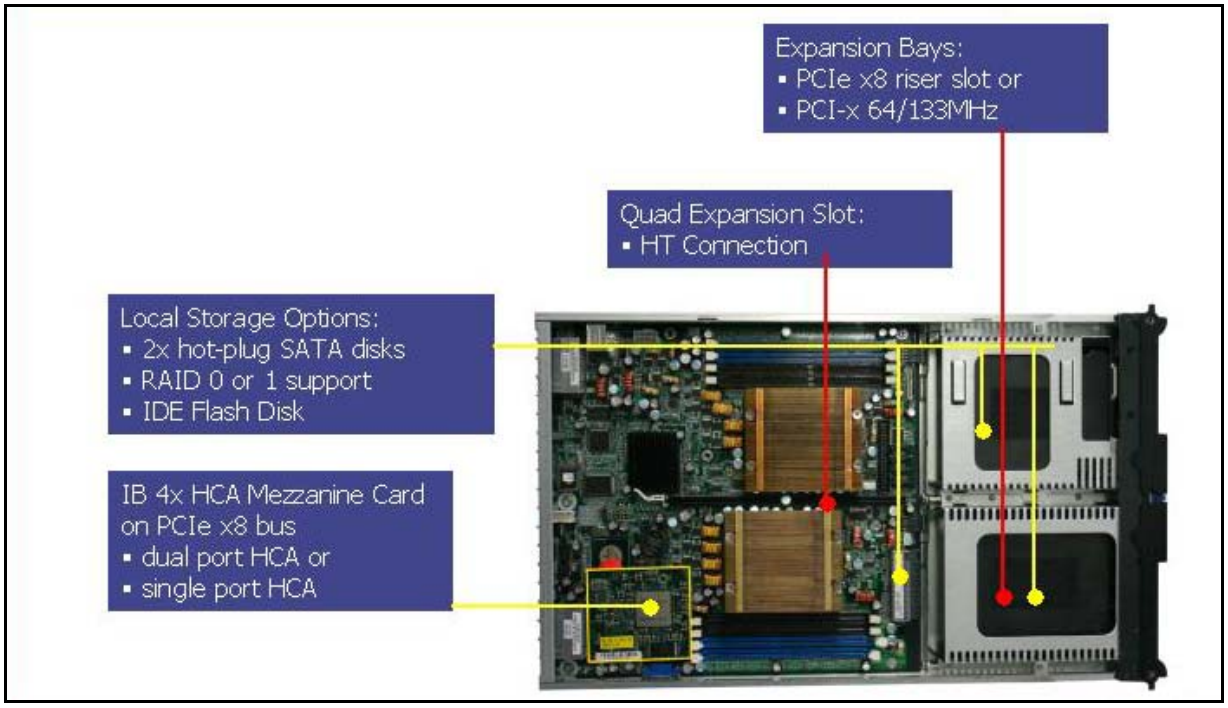

Figure 9. Xtreme Blade compute node

The CPUs are located in the middle, and the memory slots are next to them. For I/O, the InfiniBand silicon is located in the mezzanine. In the front, there is room for two hot disks, and one is drive-base modularized. This means that instead of a hot disk, there is room for a PCI Express card or a PCI-x card. This is a path to expandability in and out of the box. There is also a HyperTransport connector.

On the motherboard, the AMD Opteron™ processors attach directly to the memory. A HyperTransport connects the CPUs together. Another HyperTransport connects to a PCI Express bridge. The nVidia nForce™ 2200 (CK804) has 20 PCI Express lanes. These 20 PCI lanes are separated into two groups. One group goes to a PCI Express by eight slot. This slot allows customers to expand by putting a card in the front. It is also possible to put in a PCI-x card. The eight other PCIe lanes are connected to the Mellanox InfiniHost™ III —this is how InfiniBand is enabled on the Xtreme Blade product. The InfiniHost III is connected to the mid-plane. From the mid-plane, the signal is routed to the InfiniBand Switch Blade which utilizes a Mellanox InfiniScale™ switch product. Cables are used to connect switches. This is how the Appro Xtreme Blade is clustered using Infiniband.

A Broadcom dual-port chip 5715 is connected to the other lanes. This means that each motherboard has two gigabit Ethernet connections on the motherboard. The signals also run out the mid-plane, out to a switch, and outside the rack. The Intelligent Platform Management Interface (IPMI) protocol is an open standard and is used for remote server management. The IPMI sideband runs on the Basement Management Controller (BMC) and allows the system administrator to manage and control the server, regardless of its state. For example, if the server or the kernel crashes or is not responsive, it still is possible to reach the node through the BMC reboot or reset the server or mark it offline for temperature or fan speeds problems.

A double-width blade supports up to four CPUs in the Xtreme Blade. This provides the ability to produce a four-socket blade server with up to eight cores.

The Single-Width Blade

The Appro Xtreme Blade Server is available as a single-width blade or a dual-width blade. The single-width blade is shown in Figure 10.

Figure 10. Appro Xtreme Blade Server—single-width blade

The single-width blade provides the following features:

- Dual sockets with dual cores
- 5.2 gigaflop per core with 2.6 GHz clock
- Memory bandwidth of 2X5.9 gigabytes per second or 11.8 gigabytes per second (single core bandwidth ratio is 1.12 gigabytes/second per gigaflop/second)
- Dual GBE bandwidth is four gigabytes per second full duplex (bandwidth is 190 gigabytes/second per gigaflop/second)
- Dual SDR 4X InfiniBand is three gigabytes/second full duplex (fabric interface bandwidth is 0.14 gigabytes/second per gigaflop)

The Xtreme Blade produces about 21 gigaflops peak capability. The most challenging problems are likely to use 80% of this capability.

The memory bandwidth for both cores is about 0.56 gigabytes/second per gigaflop/second. This is slightly lower than the ideal, but there is also an option of running only one core on problems that are very memory-intensive. This makes it possible to address the best of both worlds.

For example, instead of increasing the number of processors from 100 to 400, half of the CPUs are sacrificed in order to get higher memory bandwidth for the remaining processors. Using 200 single core processors results in processors that run faster and solve the problem more effectively.

The Xtreme Blade has dual gigabit Ethernet connections. It has dual single data rate and InfiniBand connections. This unit can function as a server. It has two disk slots, but it is a limited server because of the number of drives. The Xtreme Blade can also host interface cards internally, but it can be used as a server with just a bunch of disks (JBOD) or a DataDirect storage attached.

There are two gigabit Ethernet switches, which provide redundancy for management network, shown in Figure 11.

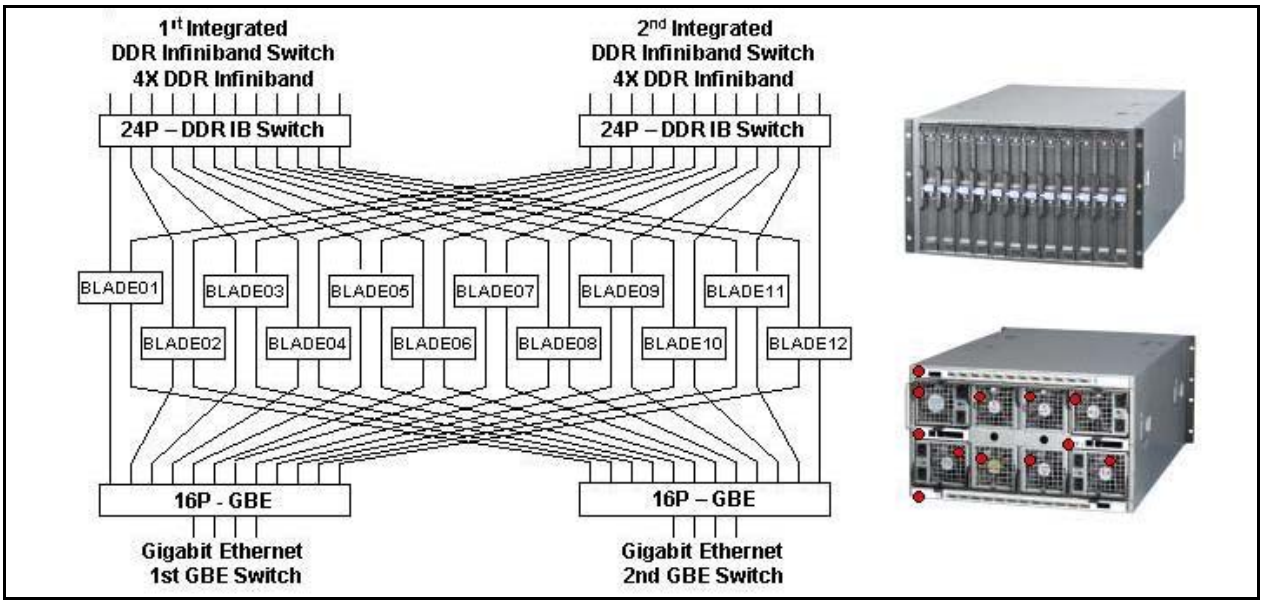

Figure 11. The Appro Xtreme Blade Server

The Dual-Width Blade

A dual-width blade is also available, illustrated in Figure 12.

Figure 12. Appro Xtreme Blade Server—dual-width blade

The dual-width blade provides the following features:

- Quad sockets with dual cores
- 5.2 gigaflop per core with 2.6 GHz clock
- Memory bandwidth of 4X5.9 gigabytes per second or 23.6 gigabytes per second (single core bandwidth ratio is 1.12 gigabytes/second per gigaflop/second)
- Dual GBE bandwidth is four gigabytes per second full duplex (bandwidth is 95 gigabytes/second per gigaflop/second)
- Dual SDR 4X InfiniBand is three gigabytes/second full duplex (fabric interface bandwidth is 0.071 gigabytes/second per gigaflop)

Only six dual-width blades will fit in a rack, but the total gigaflop capability is the same. Memory bandwidth remains the same because the number of memories is doubled. The dual-width blade has the same dual gigabit interfaces and dual fibre channel interfaces. It can function as a server because it is possible to build a 3+P RAID in it with about 1.5TB of storage. The advantages of the dual-width blade are the ability to use it as a large SMP node for capacity systems and as a server node for providing low cost secondary storage with the same blade server components. The reliability problem associated with

this type of blade server can be eliminated by using a file system such as the Teragrid High Availability File System that RAIDs the servers by combining them into parity groups and turns the network into a RAID storage system with XOR bandwidth that scales. This type of secondary storage configuration can scale with processing requirements and provide individual nodes with up to 600MB per second of secondary storage bandwidth.

The Software: ClusterStack Pro

Appro's ClusterStack Pro is based on Raytheon's Software reference design to work specifically with the Xtreme Blade solution. This hardware and software combination provides high availability, reliability, and scalability to large deployment clusters. The ClusterStack Pro is standards-based, integrated software that supports many versions of the Linux operating system. In fact, it is possible to have several versions of Linux running concurrently. Every cluster in the system could have a different version of the operating system, with another version running on the management servers.

Appro provides the fabric management (dual-rail 2D mesh), as well as resource management and scheduling. The scheduler is topology-aware. In other words, the scheduler will attempt to rearrange the job to fit into the array.

The cluster file system recommended for use with the ClusterStack Pro is the Terrascale system. ClusterStack Pro provides support for standard compilers, libraries, and utilities. It also provides MPI libraries and an IP stack with extensions for dual-rail fabric, load sharing, and fail-over. It uses a high availability fault-tolerant global file system in which both the disk and the servers are RAIDed.

ClusterStack Pro runs on duplexed servers, and its Management's RAS function owns and manages the hardware. This includes powering up, powering down, resetting, and status collection. Its Subnet Manager controls the fabric, and the routing function controls the mesh while taking care of configuration and recovery. The ClusterStack Pro also partitions the system into virtual arrays or "clusters." Each cluster has its own build of the operating system and its own scheduler. These clusters all boot from the global file system, which aggregates the disk into RAID5 arrays. It also aggregates the servers into arrays and aggregates everything into one file system. The clients have high bandwidth access to the global file system.

Using Minimization to Improve Bandwidth Only

If an organization wants to improve bandwidth only and not redundancy, you can achieve this through minimization, as shown in Figure 13.

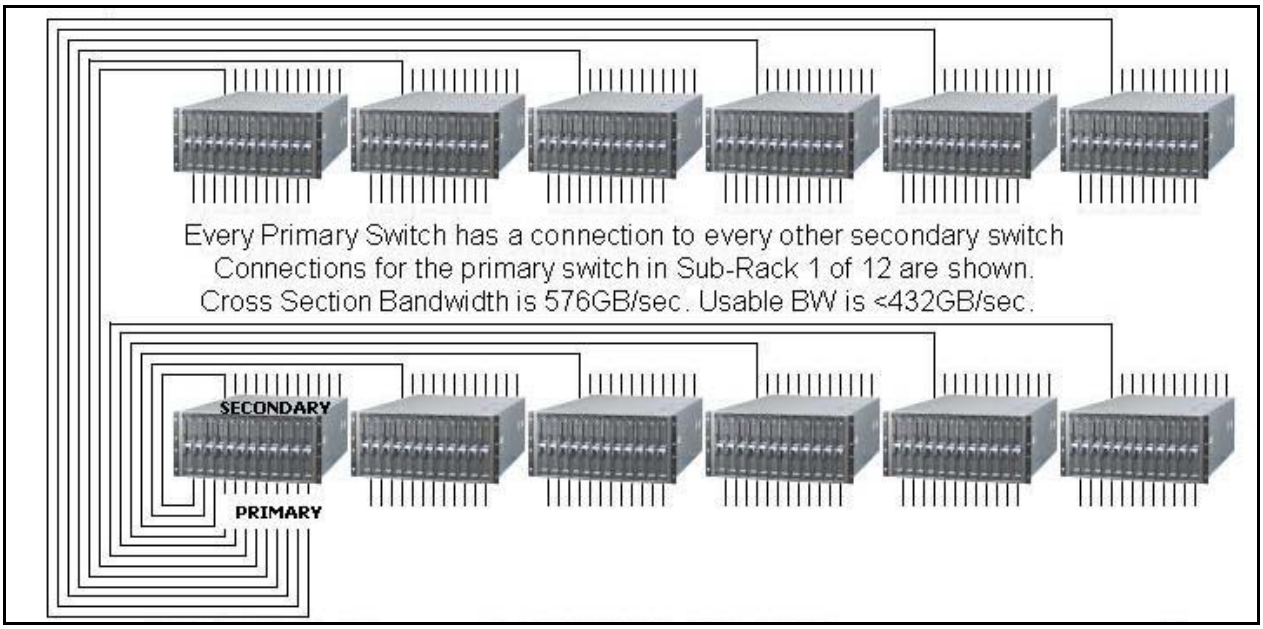

Figure 13. Twelve sub-racks/minimized fat-tree interconnect

Each edge switch functions as a core switch on the opposite side. If there is a failure, there is no redundancy but this solution provides twice the bandwidth. The main advantage of this configuration when using the Appro Xtreme Blade sub-rack is that no external switches are required. Every primary switch has a connection to every other secondary switch, and the cross-section bandwidth is the same as any fat-tree with a half bandwidth core switch. This means a system with 12 sub racks and 576 processors can be built with no external switches. In other words, all the parts can be obtained from one source. Figure 14 illustrates what such a system would look like.

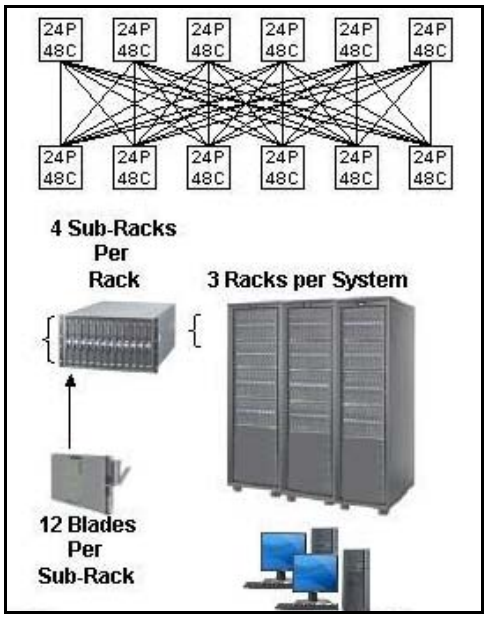

Figure 14. Twelve sub-racks/minimized fat-tree interconnect

The 12 sub-racks fat-tree interconnect includes the following:

- Three equipment racks with six sub-racks each
- Dual-rail fat-tree with DDR switch-switch connections
- Two management nodes
- Two log-on nodes
- Dual-rail fabric with minimized switch configuration
- Peak capability is 2.5 teraflop/second with 2.4 GHz AMD Opteron™ 270HE Processing Elements
- Usable capability is about two teraflop/second
- The file system's aggregate server bandwidth is 2.5 gigabyte per second and maximum client bandwidth is approximately 400 MB per second
- Maximum power required is 36 KW
- Software used is Appro's ClusterStack Pro and Terrascale's HA Global File System Software

There are four sub-racks per equipment rack and a total of three equipment racks. The sub-racks require about three kilowatts each. If an organization has power or cooling constraints it may not a good idea to fill the racks to capacity if high power processors are used.

This strategy provides some extra space for secondary storage. In this case, it is possible to have 128 diskless compute server blades with 512 compute server processing elements. Diskless nodes are another integrated feature of the ClusterStack Pro software. It is not a good idea to locate the disk in the compute nodes, as discussed earlier. Disks are more reliable outside the computer nodes. Also, if a disk is located in a compute node, it provides only 40-60 megabytes per second of bandwidth. When a compute node fails, it is not possible to use data stored on the compute node to restart the process. In this configuration, a compute node can access secondary storage at more than 400MB per second, and recovery is possible because the storage is external.

Relocating Disks to Improve Reliability

A better strategy to accomplish the same goal is to relocate disks outside the compute blades in RAID arrays to improve reliability. The InfiniBand interconnect can provide up to 400 megabytes per second back to any single compute node.

If a system is configured with 12 sub-racks, it is possible to have 128 diskless compute server blades and use 12 blades for storage server blades. The storage server blades can be configured with Terrascale's High Availability (HA) global file system software into two 4+Parity+Spare configurations. Using two server arrays with a 12-drive JBOD on each of the 12 server blades yields a storage server bandwidth of about 250MB per second per server or an aggregate of 2.5GB per second for the 12 servers.

In this example the storage server capacity is about 3.5 terabytes per server, with the disks RAIDed within the servers. The servers are then RAIDed by the software running on the clients. Therefore, if any one of the servers goes down, the spare takes its place and is rebuilt on-the-fly using data from the remaining servers. In other words, the problem is solved by software and inexpensive commodity servers. The Dual-rail network fabric minimizes the impact of having the storage connected to the fabric. This does not provide redundancy—it just improves speed.

Peak capability is about 2.5 teraflops, while usable capability is about two teraflops. Appro's ClusterStack Pro takes care of starting up, provisioning, partitioning, and scheduling the system. Raytheon uses Terrascale software, which aggregates the entire file system and provides a bandwidth of 10 servers and a capacity of eight servers. Only data servers count as capacity and the parity server counts as bandwidth, but the spare is a spare. The aggregate server bandwidth is about 2.5 gigabytes per second, which matches up with 2.5 teraflops per second because that is one megabyte per second per gigaflop per second. The

total secondary storage capacity is about 28 terabytes, which translates to about 10 bytes per flop, providing good balance. The maximum client bandwidth is about 400 megabytes per second, and the maximum power is 30 to 50 kilowatts, depending on the processors that are used.

A Better Solution for Improving Reliability

A better way to improve fabric reliability is to take the 12 connections and pair them in groups of three, which results in 4 – 12X InfiniBand connections, as shown in Figure 15.

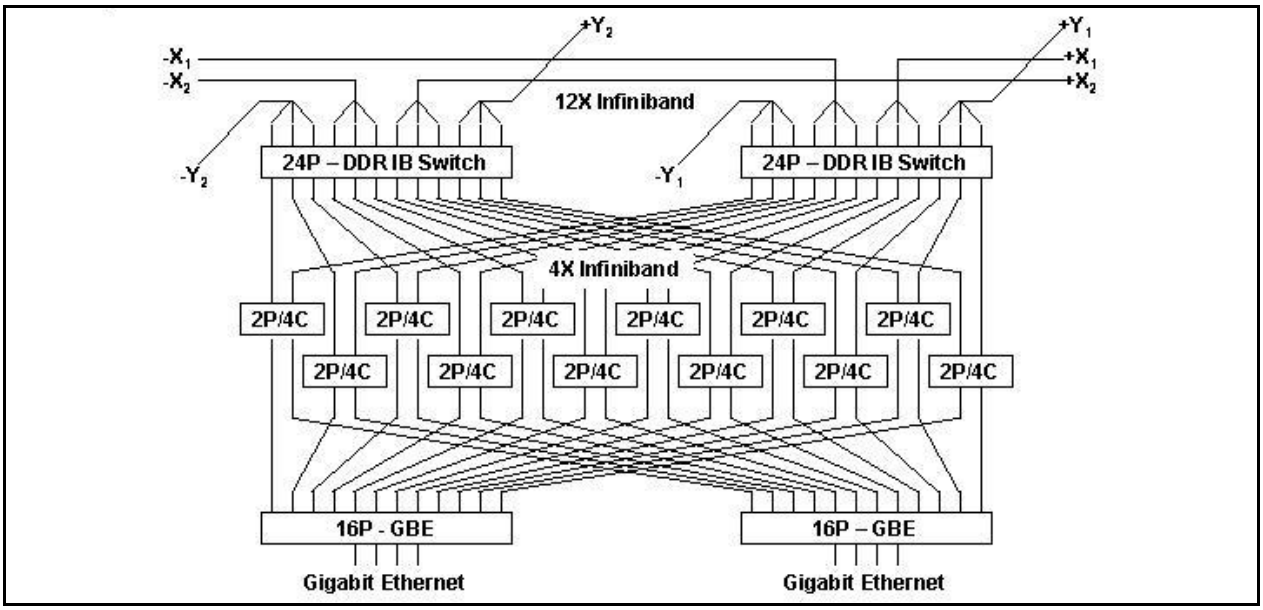

Figure 15. A better solution — 2D 12X DDR

Since the goal is to build a 2D mesh topology, it is important to achieve the highest possible performance for the connections betweens switches. By using a double data rate switch, it is possible to run single data rate 4X connections to the blades and use double data rate on the 12X connections between nodes.

Each of the switches has four X-Y connections, shown in Figure 15. There are 4–12X DDR fabric connections per switch at 12 gigabytes per second for a total full-duplex bandwidth of 24 gigabytes per second in every direction. The blades connect to both switches. The 12– 4X single data rate connections to the blades operate at an average rate of approximately 1.5 gigabytes per second. For a capacity machine a 21-gigaflop blade requires approximately 21 megabytes per second of I/O bandwidth. Achieving this requires 200 to 400 megabytes per second of peak bandwidth. In the example, the 12 blades in a sub-rack require approximately 252 megabytes per second of I/O bandwidth with a peak of perhaps two gigabytes/second.

On the other hand, a capability system requires a much higher rate of about two gigabytes per second minimum for the same 21-gigaflop blade. The peak data rate from the 12 blades in a sub-rack is approximately 18 gigabytes per second in each direction. This is much higher than the requirement for a capacity computer. Since the aggregate peak full duplex bandwidth for the blades is 36 gigabytes per second, the capability requirement for these switches would appear to be over-subscribed; however because 48 processors are being aggregated, being over-subscribed by 100 percent is about right.

Using a Dual-Rail Fabric

Figure 16 shows the dual switches that support building a dual-rail fabric, which provides redundancy and 24 gigabytes per second of bandwidth. The blades in the sub-rack connect to both switches. This fabric has four 12X DDR fabric connections at 12 gigabytes per second Full Duplex per switch and twelve 4X Single Data Rate connections to each blade at 1.5 gigabytes per second Full Duplex per switch.

Figure 16. 2D 12X DDR fabric

Using a dual-rail fabric provides high availability because if one switch or a connection fails, the MPI and IP software continue to operate at reduced bandwidth using the remaining switch. The software makes it look to the user like there is only one connection, but there are two physical connections, shown as A and B in Figure 16.

When a failure occurs in the dual-rail fabric, the software simply switches over and continues to operate on the rail that remains. In other words, one rail is not the primary that sometimes splits messages across to the other rail. In this case, both rails are equal and the load is balanced across them.

Example: Building a High Availability Single-Rack System

There are several configurations for building a supercomputer. This first example describes a high availability single-rack system. Figure 17 shows this system built with six Appro Xtreme Blade sub-racks, mounted in one cabinet and interconnected by a dual-rail 12X DDR InfiniBand fabric in a three-by-two array.

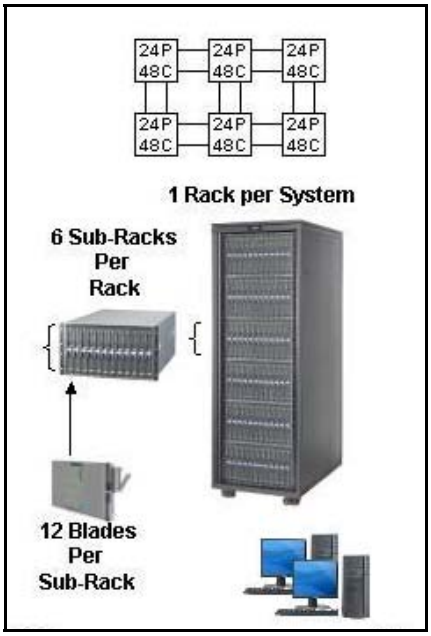

Figure 17. High availability single rack system—1.3 TF

This system has 64 diskless compute server blades with 256 compute server processing elements. Six storage server blades connect to a single DDN 9500 duplex storage control unit with 60 disk drives in 6 X (8+2) configuration that yields an aggregate server bandwidth of 1.8 gigabytes per second and 24 terabytes of storage capacity to support a 1.3 teraflop machine. This is a very well balanced configuration with more than 1 megabyte per second of secondary storage bandwidth per gigaflop and almost 18 bytes of secondary storage per flop. Again, the client bandwidth is high at 300 to 400 megabytes per second. The power required is approximately 25 kilowatts for the entire system.

There are two management servers in the system. This system has a fault-tolerant fabric and file system. It uses Appro's ClusterStack Pro and Terrascale's Global File System Software.

Example: Building a High Availability Multi-Rack System

This example shows how to take advantage of a fabric and software that are designed to scale by building the single rack system in the previous example into a high availability multi-rack system. If an organization wants a high availability multi-rack system, the solution is to hook racks together into an array as shown in Figure 18. This solution includes four equipment racks with four sub-racks each. The sub-racks are interconnected by a dual-rail 12X DDR InfiniBand fabric configured into a four-by-four array.

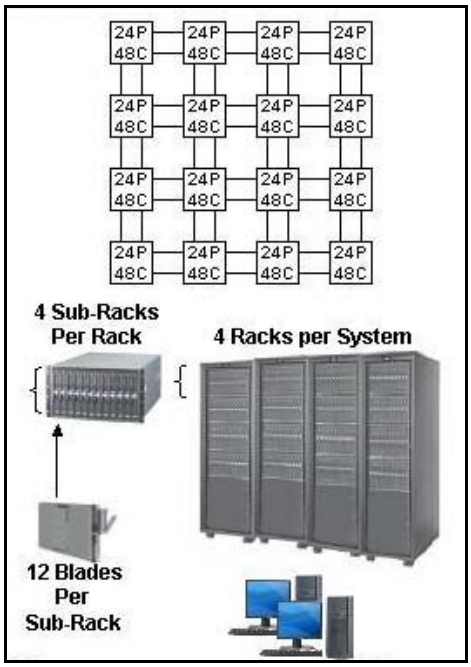

Figure 18. High availability multi-rack system—3TF

This example includes 174 diskless compute servers with 696 compute server processing elements and two DNN 9500 Duplex Control Units for storage driven by 16 storage server blades. The DDN duplex control units have a bandwidth of 2.4 gigabytes per second for each unit in this configuration. Each of the 16 storage servers is capable of delivering 300 gigabytes per second from the storage control units to the fabric. There are 80 disk drives per Duplex Control Unit. Balance is achieved with a peak capability of 3.6 teraflops per second, aggregate storage bandwidth of 4.8 gigabytes per second, maximum client bandwidth of about 400 megabytes per second, and required power of 64 kilowatts in four racks. The overall client bandwidth is much greater than can be achieved with internal disk drives, and the higher availability of the external storage arrays make a diskless configuration for the compute servers desirable.

There are two management servers with RAID1 disk arrays. Peak capability for the system is 3.6 teraflops per second with 2.6 GHz PEs. Usable capability is about 2.9 teraflops per second. This system has a fault-

tolerant fabric and file system. It uses Appro's ClusterStack Pro and Terrascale's Global File System Software running on the storage servers connected to the duplexed disk controllers.

Example: Building a High Availability Capacity System

Figure 19 illustrates a larger high availability, high capacity machine configured as a 12-by-9 array of switch nodes.

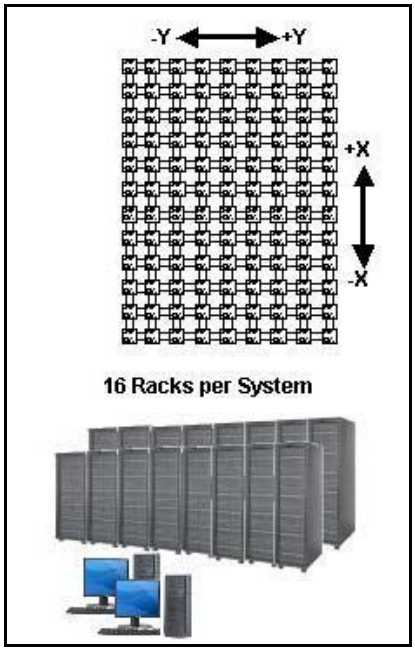

Figure 19. 21TF 12X9 array in 18 equipment racks

This system configuration has 18 equipment racks with six Appro Xtreme Blade sub-racks in each equipment rack. The sub-racks are configured into a 12-by-9 array with dual-rail X12 InfiniBand fabric connections. The system has 1024 diskless compute server blades with a total of 4,096 processing elements. The system also includes 128 storage server blades with a total of 512 processing elements. Peak capability is 21 teraflops per second with 2.6 GHz processing elements. The secondary storage system consists of 14 fault-tolerant server groups configured in 8+P RAID arrays, created by using 126 dual wide plug-in storage blades distributed across the array. The server blade disk configuration has four disk drives in a RAID5 (3 + P). Each server has a capacity of 1.5 dTB using 500 gigabyte drives and can deliver more than 167 megabytes/second using 7200RPM SATA disk drives. The server bandwidth spread across the array is 21 gigabytes per second with 168 terabytes of usable storage capacity.

The advantage of the small server building block is a fast rebuild time. When a server fails and is replaced by a spare, the data from the eight other servers in the group is used to rebuild the data on the spare unit. There are two management blades and 12 log-on or spare blades. Both compute and storage are constructed with Xtreme Blade components (single-width compute blades and double-width storage blades with four drives). This system has a fault-tolerant fabric and file system. It uses Appro's ClusterStack Pro and Terrascale's HA Global File System Software.

For the ambitious, it is possible to hook four of these arrays together with a central switch at one edge of the arrays, as shown in Figure 20.

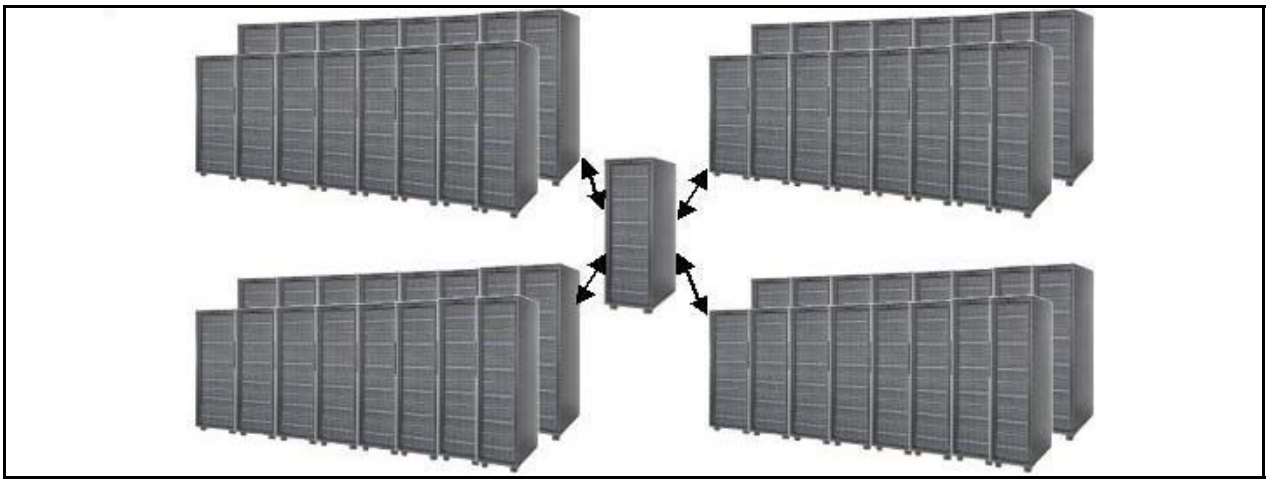

Figure 20. 84TF 4(12X9) arrays in 74 equipment racks

This approach makes all of the storage global. There is more than adequate bandwidth for capacity computing applications. Capability computing applications requiring up to half of the total array size (8192 processors) have more than 0.1 gigabytes per second per gigaflop of interconnect bandwidth. Applications using more than two quadrants have less bandwidth available between quadrants.

The core switches are located where the “Xs” are shown, near the center of Figure 21, and the arrays are arranged in a “U” shape to minimize cabling. All of the cabling is inside the cabinets with no cable being more than six meters long and most cables being less than two meters long.

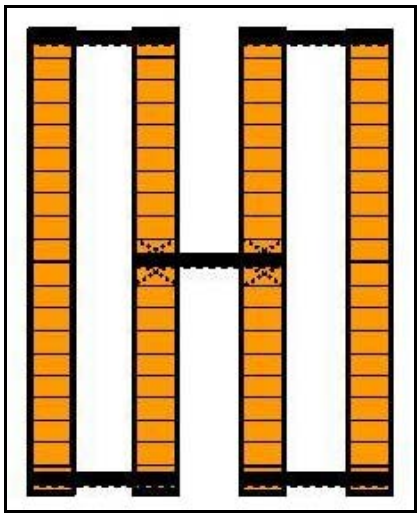

Figure 21. 84TF with integrated management and storage

The system in Figure 21 includes the following features:

- 72 equipment racks for compute and storage
- Two equipment racks for switches
- 432 Xtreme Blades sub-racks
- 4,096 dual-processor dual-core compute server blades
- 16,384 total compute server processing elements

- 21 teraflops per quadrant, 84 teraflops total
- 25 gigabytes/second per quadrant file system bandwidth and 100 gigabytes per second total bandwidth
- 128 double-width file server blades
- 512 total file server processing elements
- 14-server RAID groups in 8 + P + S configuration
- Single global file system per quadrant or total system
- 4-by-500 gigabyte disk drives in 3+P RAID array per server (double blade)
- 168dTB per quadrant file system capacity 672 dTB total
- Redundant dual-rail X12 InfiniBand fabric
- Dual-rail fabric supported by fail-over
- 1.5 megawatt power requirement
- 427 Tons Cooling requirement
- 1,600 square-foot footprint for compute and storage
- All wiring is inside or overhead
- Integrated RAS, management, and topology-aware scheduling

Conclusion

There are three goals that are always desired in building a supercomputer:

- Build the best system for the lowest cost.
- Achieve the highest possible performance.
- Achieve the highest reliability possible.

Building a state-of-the-art enterprise supercomputer requires a partnership among vendors that supply commodity parts. By partnering with Raytheon, AMD, Mellanox, DataDirect, and Terrascale, Appro combines its Xtreme Blade Server with other compatible products to provide an out-of-the-box solution that allows customers to achieve their state-of-the art supercomputing goals.