ACCRE High Performance Compute Cluster

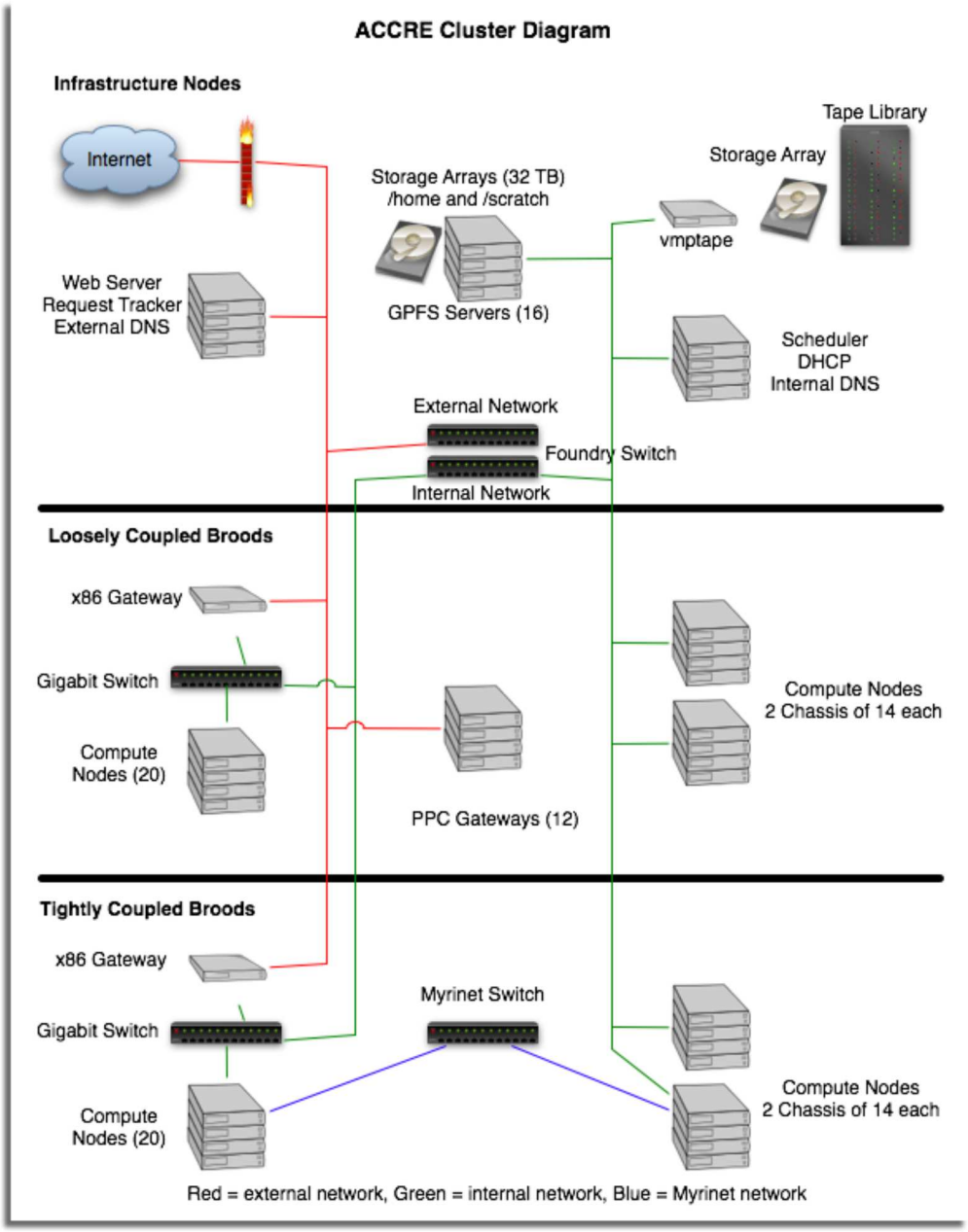
Request an Account
ACCRE Helpdesk
Utilization Charts

This page describes in detail the components and operation of the compute cluster at ACCRE. For researchers requiring text for grant proposals and publications, we also provide a less technical Summary of the ACCRE Facility.

The compute cluster currently consists of more than 600 Linux systems with either Opteron, or PowerPC dual-processors. Each node has at least 1.5 GB of memory and dual Gigabit copper ethernet ports. Over one third also have Myrinet networking.

Details of the Cluster Design

The design reflects significant input from the investigators who use it in their research and education programs. The number of nodes and the partitioning between high-performance and fast-ethernet networking is determined by demand. A schematic diagram of the system is shown below followed by a glossary of terms used in a subsequent description of the cluster:



Bandwidth -- The rate at which data can be transferred over the network. Typically this is measured in Megabits/sec (Mbps) or Gigabits/sec (Gbps).

Bi-section Bandwidth -- This is calculated by splitting the network topology in half and determining how much data can be transferred through this imaginary divider. The theoretical maximum bi-section bandwidth is half the total bandwidth of all connected computers.

Brood -- The fundamental cluster building block: either a group of 20 x86 compute nodes, plus a gateway and associated switch, or a group of 28 PowerPC blades plus a gateway.

Compute Node -- A node whose purpose is to run user applications. Users will not have direct access to these machines. Access will be granted solely through the job scheduler. A compute node may be either a 1U rack mount node or a blade node.

Disk Server -- A machine that is used for data storage. Users will not normally have access to these machines. For more information see the **description of GPFS** below.

Fast Ethernet -- Commodity networking used in most desktop computers that has a bandwidth of 100-Mbps and latency of up to 150 microseconds in Linux.

Gateway or Management Node -- Computer designed for interactive login use. Users log in to these machines for compiling, editing, and debugging of their programs and to submit jobs. There is one gateway/management node per brood.

Gigabit Ethernet -- Commodity networking typically found in servers and has a bandwidth of 1 Gbps and latency of up to 150 microseconds in Linux.

High-performance network -- Low latency, high bandwidth, and scalable network. Currently, we use Myrinet as the basis for this network.

Latency -- The amount of time required to "package" data for sending over the network.

Myrinet-- A low latency (16 microseconds), high bandwidth (2 Gbps), and scalable cost-effective network typically implemented over fibre. Myrinet has a proven track record as a capable high-performance network.

Cluster Connectivity

The ACCRE cluster has a flexible network topology which can accommodate different users and their needs, while leaving room for expansion. There are three separate functional networks: (1) external connectivity, (2) management and low bandwidth application, and (3) high-performance application.

The gateways are connected to the Internet through the external network comprised of gigabit ethernet links to our core **Foundry BigIron RX16** switch. In addition, the Foundry switch is partitioned among 2 networks, one internal, one external.

The management network is used for both data traffic, such as **GPFS**, and for health monitoring of the nodes. As a result, the management network is required to be scalable and have a high bandwidth. A combination of fast ethernet and gigabit ethernet is sufficient. The network design is a classical fat tree with all of the disk servers and management nodes connected at the top level, allowing deployment of their full bandwidth to any brood. Only a constant incremental cost per machine is incurred for each additional machine.

For the x86 broods the rest of the management network is local within a brood as can be seen from the diagram. Each brood has a local switch. This brood switch is connected to the top level switch using a 1 gigabit uplink. The compute nodes are connected to this same brood switch with fast ethernet or gigabit ethernet; the gateway node is always connected with gigabit ethernet.

For the PowerPC broods each chassis of 14 blades has a gigabit ethernet connection to the internal network portion of the core Foundry BigIron RX16 switch

Some research applications on the cluster require a high-performance connectivity between nodes that has both high bandwidth and low latency. In these cases the management network between nodes does not suffice. For this reason, a subset of the compute nodes are connected by a separate Myrinet network. Other than the addition of the Myrinet card these nodes are identical to the other compute nodes. The Myrinet network is used by the research applications with no management traffic.

Node installation, maintenance updates, and health monitoring

Initial OS installation is accomplished using the open source software **SystemImager**. We make an image of the configured and operational system, subsequently replicating this image across all nodes. Multiple images for different compute node configurations are stored on an infrastructure node. By using this technique it becomes possible to perform a wipe and re-install of the entire cluster in a short period of time. Similarly, SystemImager is also used to update a node, transferring only the data needed for the update instead of the entire image. Because of this intelligent update, most common maintenance operations, such as updating the kernel, take mere minutes.

Each x86 brood can be configured as a stand-alone mini-cluster. This mini-cluster can then be used to test software and hardware updates without interfering with the rest of the cluster.

The health of the compute nodes within a brood are monitored through the use of the open source package **Nagios** which supports distributed monitoring. Distributed monitoring allows data from individual management nodes to be collected on a single machine for viewing, analysis, and problem notification.

(Top of Page)

Detailed Node Configuration

The ACCRE Linux cluster is comprised of dual processor dual core x86 2.4 GHz and 1.8 GHz Opterons, dual processor x86 (2.0 GHz Opteron) nodes, dual processor quad core 2.4 GHz Opteron and dual processor quad core 2.3 GHz Intel Nehalem. The cluster has over 2000 processors and the theoretical peak performance is roughly 12 TFLOPS. As shown in the table below, there is a heterogenous mix of memory configurations. Gateway node names contain the letter "s".

The VAMPIRE Cluster (April 2010)**

# Compute Processors (Nodes/Blades)	Processor Speed (GHz)	Memory (GB)	Available Memory (GB)	Connectivity Type	Compute Node/Blade Names*	Primary Node Properties	Gateway Names

AMD Opteron							
308 (154 dual)	2.0	2.0	1.8	Ethernet	vmp161-vmp220 vmp301-vmp380	x86,opteron	vmps09-vmps12 vmps16-vmps20
40 (20 dual)	2.0	4.0	3.8	Ethernet	vmp381-vmp400	x86,opteron	vmps20
80 (40 dual)	2.0	2.0	1.8	Myrinet	vmp241-vmp280	x86,opteron,myrinet	vmps14
160 (40 dual dual)	2.4	4.0	3.8	Ethernet	vmp401-vmp440	x86,opteron	vmps21
80 (20 dual dual)	2.4	16.0	15.8	Ethernet	vmp001-vmp020	x86,opteron	vmps01
234 (20 dual dual)	2.4	8.0	7.8	Ethernet	vmp021-vmp040	x86,opteron	vmps01
960 (120 dual-quad)	2.4	32.0	31.8	Ethernet	vmp041-vmp160	x86,opteron	vmps04-vmps05 vmps07
Intel Nehalem							
280 (35 dual quad)	2.3	24	23.8	Ethernet	vmp502-vmp536	x86, nehalem	vmps50
96 (12 dual quad)	2.3	48	43.8	Ethernet	vmp537-vmp548	x86,nehalem	vmps50
376 (47 dual quad)	2.3	24	23.8	Ethernet	vmp522-vmp598	x86, nehalem	vmps55

** For requesting nodes with big memory, please see **this FAQ**.

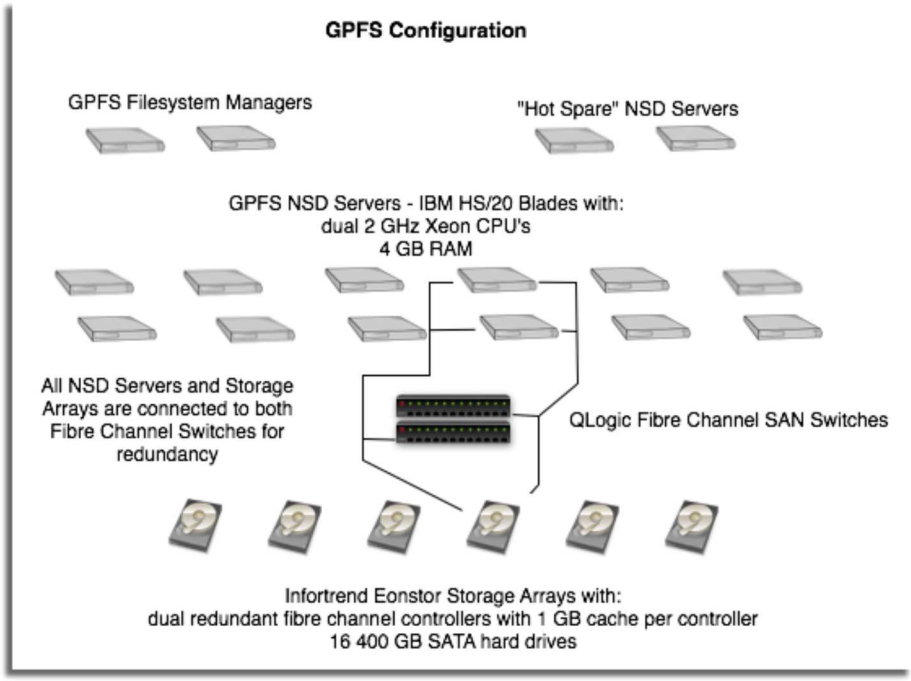
(Top of Page)

Cluster File System - GPFS

One of the fundamental challenges of building a Linux cluster is to make sure that data is available when needed to any CPU in any node in the cluster at any time. Having a cluster with the very latest CPUs from AMD, Intel, or IBM means very little if those CPUs spend much of their time idle, waiting for data to process. In small clusters data is typically made available to the nodes via a Network File System (NFS) server, which stores all of the user data and exports it to the entire cluster. Applications can then access data as if it were local. However, there are two significant disadvantages to this approach. First, the NFS server is a single point of failure. If the NFS server is unavailable for any reason then the entire cluster is unavailable for use and jobs which may have been running for weeks may be killed. This single point of failure can be eliminated by clustering two (or more) servers, but this can quickly become very costly and, more importantly, does not solve the second significant disadvantage of NFS: poor performance. NFS typically does not scale to much over 100 MB/second bandwidth. Because of this ceiling, performance may be adequate for small clusters of up to approximately 150 nodes; scaling to a greater number of nodes causes the limitations of NFS to quickly become apparent.

Because the ACCRE cluster began as a small (120 nodes) cluster, ACCRE initially took the NFS server approach. However, by the fall of 2004 both of the limitations of the NFS server approach had become apparent and a search for a more robust, higher performance solution was initiated. After a lengthy evaluation process, ACCRE selected **IBM's GPFS (General Parallel Filesystem)** to replace NFS. GPFS was placed into production in August of 2005.

In a GPFS cluster two servers are set up as GPFS filesystem managers. In addition, two or more servers are set up as disk I/O servers called NSD (Network Shared Disk) servers. NSD servers are connected to one or more storage arrays in a redundant configuration. This configuration may be accomplished by directly connecting two NSD servers to a single storage array with dual controllers or by connecting multiple NSD servers to multiple storage arrays via a SAN (Storage Area Network). ACCRE initially planned to set up GPFS using a SAN configuration. However, issues with OS/driver compatibility forced us to set up 10 NSD servers directly connected in pairs to 5 storage arrays. The home and scratch filesystems were then created with data and metadata striped across multiple NSD servers/storage arrays. In early 2006, IBM resolved the OS/driver compatibility issue and ACCRE switched to a SAN configuration during a cluster downtime in August of 2006. During this downtime we also added a 6th storage array configured specifically for and dedicated to storing metadata. All metadata was moved off the 5 existing storage arrays to the new storage array. This has enhanced the performance of metadata related operations (for example, "ls") and data I/O (since they are not impacted by someone doing an "ls"). We subsequently added a 7th storage array and mirrored the metadata from the original metadata storage array. This provides both redundancy and additional performance. Since each of the NSD servers has dual bonded gigabit ethernet connections to our core Foundry switch, our GPFS cluster is capable of sustaining 2 GByte/second data transfer to the compute nodes.



In March of 2008 we upgraded to a new major revision of GPFS. The main benefit of this upgrade is that it allows us to specify up to 8 NSD servers for each GPFS disk in the storage arrays. This has already prevented a cluster outage when a power outage caused multiple NSD servers to go down simultaneously.

However, we have experienced outages due to the way GPFS is implemented and used at Vanderbilt. For example, most clusters using GPFS have one type of hardware for their compute nodes (we have three: Opteron, dual-core Opteron, and PowerPC) and run the same operating system on all of them. In addition, the way in which the majority of user applications access GPFS at Vanderbilt is not typical. Most large clusters using GPFS run a few very large multi-processor applications which either do massive amounts of I/O in parallel to a single large data file or do virtually no I/O at all. At Vanderbilt, the cluster is used primarily to run very large numbers of single-processor jobs which do very small I/O to large numbers of very small files. This causes various contention issues not seen in more "typical" clusters.

In spite of the issues we have encountered, we believe that GPFS is still the best option currently available to us. Going back to NFS would be a step backwards. None of the other cluster filesystems currently available offer any significant advantages over GPFS. GPFS is used by many of the fastest supercomputers in the world as ranked by the **Top 500 list**. We will continue to evaluate all options available to us in order to provide the most reliable, highest performing filesystem available to our user community.

(Top of Page)

Data Storage and Backup

The **True incremental Backup System (TiBS)** is used to backup the ACCRE cluster home directories nightly. Currently the Quantum ATL P7000 Tape Library is used for the cluster disk backup. A big advantage of TiBS is that it minimizes the time (and network resources) required for backups, even full backups. After the initial full backup, TiBS only takes incremental backups from the client. To create full backups, an incremental backup is taken from the client. Then, on the server side, all incrementals since the last full backup are merged into the previous full backup to create a new full backup. This takes the load off the client machine and network. The integrity of the previous full backup is also verified. Please see our **disk quotas and backups policies** for more information. (TiBS is available for all current operating systems and apart from the cluster, ACCRE also offers backup services for data located remotely. This service is through special arrangement. If you are interested, please see our **Tape Backup Services** and contact **ACCRE Administration** for more details.)

(Top of Page)

Resource Allocation

A central issue in sharing a resource, such as the cluster, is making sure that each group is able to receive their fairshare if they are regularly submitting jobs to the cluster, that groups do not

interfere with the work of other groups, and that research at Vanderbilt University is optimized by not wasting compute cycles. Resource management, scheduling of jobs, and tracking usage is handled by the **Moab Scheduler** and **TORQUE Resource Manager**

PBS is used to provide low level resource management. It supplies user functionality to submit jobs and to check system status. It is also used by Moab to start and stop jobs, to collect job output, and to return output to the user. TORQUE allows users to specify attributes about the nodes required to run a given job, for example Myrinet versus fast ethernet.

Moab is a flexible job scheduler designed to guarantee, on average, that each group or user has the use of the particular number of nodes they are entitled to. If there are competing jobs, processing time is allocated by calculating a priority based mainly on the “fair share” mechanism of Moab. On the other hand, if no jobs from other groups are in the queue it is possible for an individual user or group to use a significant portion of the cluster. This maximizes cluster usage while maintaining an equitable sharing.

For specific details about ACCRE resource allocation, please see more about **ACCRE job scheduler parameters**.

(Top of Page)

Installed Applications

The cluster offers GCC, Intel, and Absoft compilers, with support for PVM, MPICH, MPICH-GM, FFTW, LAPACK, BLAS, GSL, Dakota, R, Matlab, and several other packages/libraries. Please browse a **comprehensive list of research computing software products**.

(Top of Page)

Last modified: March 29 2010 05:05:49 CST.