Introduction to the Cluster Grid – Part 1

James Coomer, Solutions Technology Group, UK

Charu Chaubal, Grid Computing

Sun BluePrints™ OnLine—August 2002

Introduction to the Cluster Grid – Part 1

This article is an introduction to the Sun Cluster Grid. The first section of this article provides a generic technical description of basic cluster grid architecture. The second section describes the functionality and architecture of the software components of Sun's Cluster Grid stack.

This article is intended for IT professionals, system administrators, and anyone interested in understanding the concepts of a cluster grid.

This article is as a primer for a subsequent Sun BluePrints™ OnLine article entitled "Introduction to the Cluster Grid – Part 2", which discusses the process of designing and implementing a Sun Cluster Grid.

What is Grid Computing?

Traditionally, grid computing only existed in the realms of high performance computing and technical compute farms. Increasing demand for compute power and data sharing, combined with technological advances, particularly relating to network bandwidth increases, has extended the scope of the grid beyond its traditional bounds.

Grid computing provides an environment in which network resources are virtualized to enable a utility model of computing. Grid computing provides highly available services with a high degree of transparency to users. These services can be delivered through Quality of Service guarantees, auto-negotiated contracts, metered access, and so forth.

Types of Grid Environments

Grid computing can be divided into the following three logical levels of deployment:

■ Global grids
■ Enterprise grids
■ Cluster grids

Global grids are collections of enterprise and cluster grids as well as other geographically distributed resources, all of which have agreed upon global usage policies and protocols to enable resource sharing.

Enterprise grids enable multiple projects or departments to share resources within an enterprise or campus, and don't necessarily have to address the security and global policy management issues associated with global grids.

A cluster grid is the simplest form of a grid, and provides a compute service to the group or department level. The class of software that enables this service might be a distributed resource management (DRM) system, job management system (JMS), or job scheduling system. The different terms hint at the extent of the functionality of the software. For example, a DRM system might have features beyond those of a job scheduling system such as catering for heterogeneous, distributed architectures. This article describes the cluster grid in terms of a DRM system.

The key benefit of cluster grid architecture is to maximize the use of compute resources, and increase throughput for user jobs. To achieve this, DRM software provides a virtualization of the compute resource.

In addition to the DRM software, cluster grids typically have system management software to collectively monitor and administer the infrastructure, and tools to facilitate application development for the grid.

The cluster grid is a superset of other technical compute resources such as Linux clusters, throughput clusters, midrange compute servers, and high-end shared-memory systems. As such, the cluster grid can operate within a heterogeneous environment with mixed server types, mixed operating environments, and mixed workloads.

A cluster grid can be implemented as a convenient provision of departmental compute resources, or it can be a precursor to the development of an enterprise grid, where multi-departmental access is desired. In addition, a cluster grid can be integrated with middleware, such as that developed by the Avaki Corporation and the Globus Project, to provide a compute service to a higher level grid.

Cluster Grid Architecture

The cluster grid architecture is divided into the following three nonhierarchical, logical tiers:

- Access
- Management
- Compute

Each tier, shown in FIGURE 1, is defined by the services it provides. The access layer provides the means to access the cluster grid for job submission, administration and so on. The management layer provides the major cluster grid services such as job management, health monitoring, NFS, and so on. The compute layer provides the compute power for the cluster grid, and supports the runtime environments for user applications.

Each layer can be considered independently to some extent. The three-tier definition enables consideration of the sizing, scalability, and availability, of each tier separately.
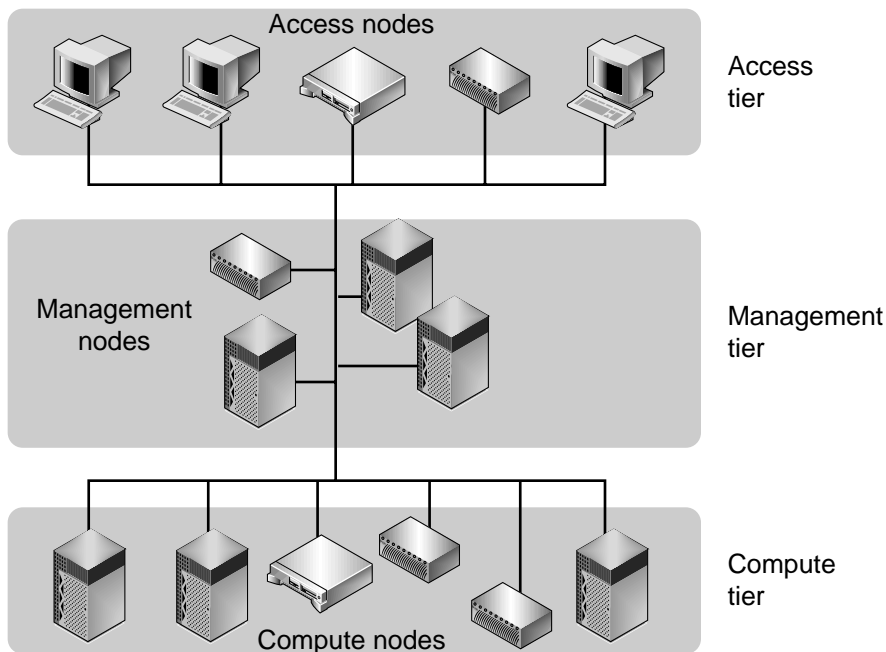
FIGURE 1   Three Tiers of the Cluster Grid Architecture

Access Tier

The access tier provides access and authentication services to the cluster grid users. Conventional access commands such as `telnet`, `rlogin`, `ftp,` and `ssh`, can be used to access the system. Web-based services can be provided to permit easy or tightly controlled access to the facility. Beyond the ability to configure, submit, and control compute jobs, web-based services provide accounting information or administrative functions. Any access method should be able to integrate with common authentication schemes such as NIS or LDAP.

The access tier can be enhanced for integration with a global grid. For example, the Globus Resource Allocation Management software of the Globus 2.0Toolkit implements a gatekeeper process that performs authentication for grid users, and allows job submission to the Sun™ Grid Engine (described later) through a job manager process.

Management Tier

The middle tier is responsible for providing the major cluster grid services: DRM, hardware diagnosis software, system performance monitoring, and so on. While the DRM is a required feature of a cluster grid, additional duties of this tier can also include:

- File service—Provides file-sharing service for user home directories, libraries, compilers, applications, etc.
- License key service—Manages software license keys, such as compiler licenses for the cluster grid.
- Backup Management—Provides traditional or hierarchical storage management services.
- Install service—Manages operating system and application software versioning, and patch application on other nodes in the cluster grid.

The size and number of servers in this tier vary depending on the type and level of services required. For small implementations with limited functionality, a single node hosts all management services for ease of administration. Alternatively, multiple servers provide these functions for greater scalability, flexibility, and availability.

Compute Tier

The compute tier supplies the compute power for the cluster grid. Jobs submitted through upper tiers in the architecture are scheduled to run on one or more nodes in the compute tier. Nodes in the compute tier run the client- or agent-side processes of the DRM software, daemons associated with message-passing environments (for multiprocessing), and agents for system health monitoring. The compute tier communicates with the management tier, receiving jobs, and reporting job completion status and accounting details.

The compute tier can be heterogeneous in terms of several characteristics as follows:

- Servers—The hardware characteristics can be radically different across a cluster grid. Machines can be symmetric multiprocessing (SMP) or uniprocessor, with differing physical memory sizes, CPU cache sizes, and so on.
- Platform—Hosts might run the Solaris™ operating environment, Linux, Aix, Irix, or other operating systems on different processor architectures such as SPARC™, Intel, Alpha, and others.
- Function—Groups of nodes in the compute tier perform different functions, and support several functions, or change functions based on a calendar. Functions include interactive, batch, visualization, and parallel.
- Interconnect—Some compute hosts can be networked through a specialized low-latency interconnect, such as Myrinet, or Sun Fire™ Link.

The DRM software coordinates job requests with the appropriate compute hosts, taking these characteristics into account.

Cluster Grids on Sun Hardware

You can use Sun supported or Sun open-source software to implement cluster grids on Sun systems running the Solaris or Linux operating environments.

You gain the following features and benefits that a cluster grid offers:

- Maximizes resource availability and usage—dynamically aggregates the compute power of all systems in the grid (including individual subsystems such as CPUs, RAM, storage, availability, and so on), providing total available compute power to the grid.
- Automates resource management—analyzes large numbers of compute jobs and distributes them to the most appropriate resource in the grid, while allowing for configurable job prioritization and scheduling.
- Transparent user access to available resources.
- Provides centralized administration—administrators control resource usage through a single centralized interface.
- Full 64-bit support for the hardware and software processes (Solaris OE only).
- Support for single-threaded, multi-threaded, and multiprocessor applications
- Make the most of existing Sun systems in your environment, from thin clients and individual workstations to powerful servers.
- Scalability because the architecture supports many thousands of processors, with optimized parallel messaging for minimum network load.
- Low cost implementation—All the components of the cluster grid stack are available at no cost with Sun supplied hardware. Both Sun Grid Engine software and Grid Engine source code are available at no cost. HPC ClusterTools technology source code and SunVTS are also freely available for download. See "Related Resources" on page 16 for details.

Implementing a cluster grid with Sun supported open-source software involves installing the Sun Cluster Grid software stack on one or more systems.

Sun Cluster Grid Software Stack

The Sun Cluster Grid software stack comprises a number of software components:

- Sun Grid Engine (SGE) software
- Sun HPC ClusterTools™ software

In a Solaris operating environment, the following tools provide additional features to the cluster grid environment:

- Solaris JumpStart™ software
- Solaris™ Flash software
- Sun™ Management Center (SunMC) software
- Sun Validation Test Suite (SunVTS™) software

The single essential component of a Sun Cluster Grid is the Sun Grid Engine (SGE) software. Implemented alone, the SGE software delivers job management services for Solaris and Linux environments. The other components of the Sun Cluster Grid stack provide additional functions as shown in FIGURE 2. Brief details are given in this article on the function and architecture of each component.

The components are available at no cost, and some are available as open-source software (see "Related Resources" on page 16).

The block diagram in FIGURE 2 shows the different levels of implementation of the Sun Cluster Grid software stack as follows:

- Left stack—Represents a basic stack, with only the SGE software installed as the DRM.
- Center stack—Includes the addition of the Sun HPC ClusterTools software component for support of Sun message passing interface (MPI) applications (a standard for message-passing libraries), and development tools with runtime libraries.
- Right stack—Implements the full compliment of components, including the system management tools (SunMC and SunVTS software) for health monitoring and hardware testing.
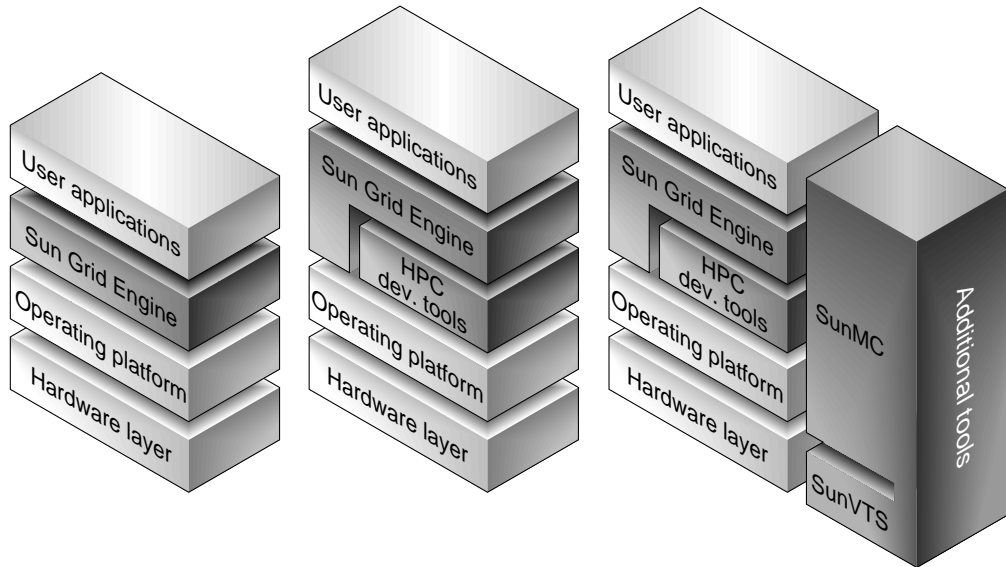
FIGURE 2   Sun Cluster Grid Software Stack

> **Note –** The Sun Grid Engine software is supported on the Solaris and Linux operating environments, while the other stack components are only supported on the Solaris operating environment. A wide variety of platforms, however, can be controlled by the Sun Grid Engine through the use of unsupported binaries which are available from the open-source site.

Sun Grid Engine

The SGE distributed resource management (DRM) software is the core component of the Sun Cluster Grid software stack. The SGE software provides all of the traditional DRM functions, such as batch queuing, load balancing, job accounting statistics, user-specifiable resources, suspending and resuming jobs, and cluster-wide resources. The SGE software also includes enhancements such as a batch-aware shell, Qtsch, which allows interactive applications to be used with the SGE software.

The Sun Grid Engine, Enterprise Edition (SGEEE) software delivers additional capability beyond that of the SGE software in the form of a policy module. Apart from the policy module, the SGEEE and SGE software are largely identical. The policy module lets administrators allocate a share of a compute resource to different departments and projects, and is therefore considered the DRM software of choice for an enterprise grid rather than the cluster grid. The SGEEE software is offered free in the form of open-source software called Grid Engine (GE).

There are four types of logical hosts in an SGE environment, as shown in FIGURE 3 and described in the following table. Depending on the size, complexity, and desired scalability of the cluster grid, a grid can be set up with one or two systems with multiple host roles. In fact, you can set up a grid with only one or two systems, each configured with multiple host roles.
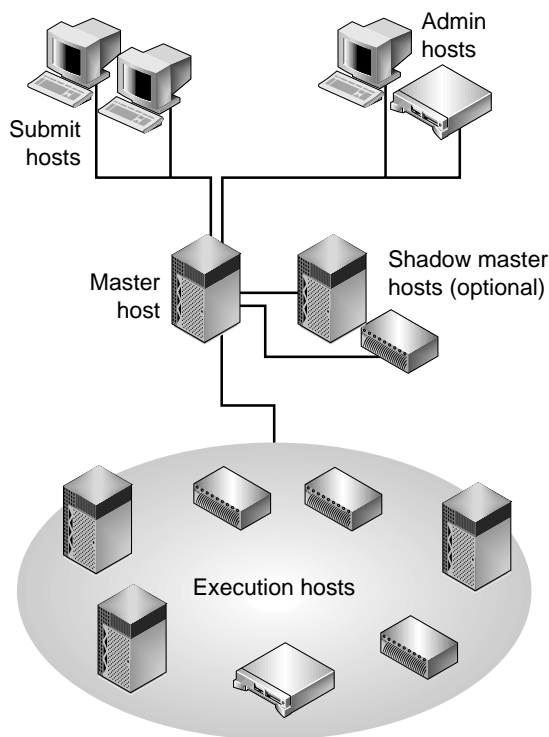
FIGURE 3   Hosts in a Sun Grid Engine

TABLE 1   Hosts in a Sun Grid Engine

| Host	Description
Master	The host that handles all requests from users, makes job scheduling decisions, and dispatches jobs to execution hosts. There is a single master host in each SGE software implementation. Daemons: • Schedd—Is the scheduler daemon that matches jobs in the spooling area to available hosts, depending upon job priority, job requirements, etc. • Qmaster—Accepts job requests and passes them on to the scheduler daemon, and implements the scheduling decisions made by the scheduler daemon.
Shadow master	While there is only one master host, other machines in the cluster can be designated as shadow master hosts to provide greater availability. A shadow master host continually monitors the master host, and automatically and transparently assumes control in the event that the master host fails. Daemon: • Shadowd—Monitors the existence of the master host daemons, and arranges to take over those functions if the master host fails.

TABLE 1   Hosts in a Sun Grid Engine

Host	Description
Execution	Hosts in the cluster that are available to execute jobs are called execution hosts. Daemon: • `Execd`—Accepts jobs from the `qmaster` daemon and spawns a shepherd process to execute the job on the local machine. Reports load information back to the master daemon.
Submit	Hosts configured to submit, monitor, and administer jobs. No daemons are required on submit hosts.
Administration	Hosts used to make changes to the cluster configuration, such as changing DRM parameters, adding new nodes, or adding or changing users. No daemons are required for administration hosts.

Software Job Flow

All jobs are submitted to the SGE master and are held in a spooling area until the scheduler determines that the job is ready to run. The SGE software matches available resources to the job requirements, such as physical memory, CPU speed, and software license needs. As soon as an appropriate resource becomes available for execution of a new job, the SGE software dispatches a matching job with the highest priority. The SGE scheduler takes into account the order the job was submitted, what machines are available, and the priority of the job.

The following description and figure illustrates a typical software job flow.

1. Job submission—When a user submits a job from a submit host, the job submission request is sent to the master host.

2. Job scheduling—The master host determines the host to which the job will be assigned. It assesses the load, checks for licenses, and evaluates any other job requirements.

3. Job execution—After obtaining scheduling information, the master host then sends the job to the selected execution host. The execution host saves the job in a job information database and starts a shepherd process, which starts the job, and waits for completion.

4. Accounting information—When the job is complete, the shepherd process returns the job information, and the execution host then reports the job completion to the master host, and removes the job from the job information database. The master host updates the job accounting database to reflect job completion.
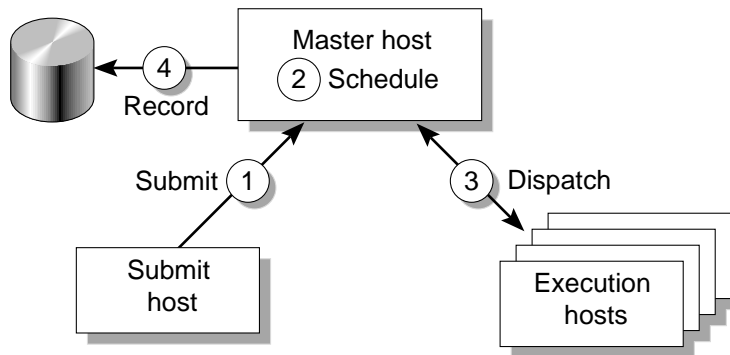
FIGURE 4   Sun Grid Engine Software Job Flow

Sun HPC ClusterTools Software

Sun HPC ClusterTools software provides an integrated software environment for MPI applications, including additional high performance libraries (Scalapack), a parallel debugger, and the runtime environment for the MPI applications.

Sun HPC ClusterTools software is thread-safe, facilitating hybrid parallel applications that mix threaded and MPI parallelism to create applications that use MPI for communication between cooperating processes and threads within each process. Such codes can make the most efficient use of the capabilities of individual SMP nodes in the high performance cluster environment.

The Sun™ Cluster Runtime Environment (CRE) provides the execution environment necessary for launching Sun MPI parallel jobs and load balancing across a compute cluster. The CRE comprises two sets of daemons—the master daemons and the nodal daemons. These two sets of daemons work cooperatively to maintain the state of the cluster and manage program execution. The master daemons consist of the daemons `tm.rdb`, `tm.mpmd`, and `tm.watchd`. They run on one node exclusively, which is called the *master node*. There are two nodal daemons, `tm.omd` and `tm.spmd`. They run on all the nodes.

Integration with the Sun Grid Engine Software

In a production computing environment, the CRE component of Sun HPC ClusterTools software can be integrated with the SGE software to handle the details of launching and controlling Sun MPI jobs. Sun CRE provides the SGE software with all the relevant information about parallel applications in which multiple resources are reserved for a single job.

Currently, the integration of Sun HPC ClusterTools software with the SGE software is "loose". Tight integration is supported with other MPI runtime environments such as MPICH (a portable MPI implementation).

Additional Tools for the Grid

Additional software tools facilitate the installation, application development, administration, and maintenance of components in the cluster grid. Unless specified otherwise, the tools mentioned in this article are distributed with the Solaris operating environment.

Solaris JumpStart and Solaris Flash Software

Solaris JumpStart software and Solaris Flash software are tools that speed and aid the automated installation of the operating environment for Sun systems. The tools can be used in tandem to manage the installations in a cluster grid. Both Solaris JumpStart software and Solaris Flash software support the use of post-installation scripts that can be executed automatically following the install. In a cluster grid environment, these scripts can be used to mount the SGE master directories and perform OS installations.

Solaris JumpStart software is an automated installation program that installs and sets up a Solaris operating environment anywhere on the network, without any user interaction. It allows the administrator to have different custom jump start configurations based on rule sets. Systems undergoing network installs are matched to a system profile using the `rules` file. The `rules` files is a look-up table consisting of rules that define matches between system attributes and profiles. Profiles contain file system layout and installation package configuration information that is used during the installation process.

The Solaris Flash software captures a snapshot image of a system, including the Solaris operating environment, most applications, and system configuration information, using the `flarcreate` command (flash tool). Using this system image, administrators can replicate server configurations on to multiple (clone) servers. The Flash archive is transmitted over the network from an NFS server, or HTTP server to the installation client, and is written directly to the client's disk. After the archive is written to the disk, any necessary configuration modifications are performed.

Sun Management Center Software

SunMC software is system management software for system administrators to perform remote system management, monitor performance, and isolate hardware and software faults for Sun systems.

SunMC software has three components; console, server, and agent. The console component provides user interface in either the SunMC GUI or through a web browser. SunMC software is based on an intelligent agent-based architecture. A SunMC server monitors and controls managed entities by sending requests to agents residing on the managed nodes. Agents are software components based on simple network management protocol (SNMP) technology that collect management data on behalf of the server. Agents collect and process data locally, and can act on data to send SNMP traps and run processes. These agents can also initiate alarms, provide administrative notification, or perform specific actions based on collected data or messages through customizable rules and thresholds.

In the cluster grid environment, SunMC agents can be installed on those servers that provide critical services such as the SGE master or NFS servers, and also for the larger compute tier hosts. For thin-node compute hosts (single or dual processor), if the agent layer is not installed, the SunMC server will still report "alive status" using SNMP pings.

Sun Validation Test Suite Software

SunVTS software is a diagnostic tool that tests and validates Sun hardware by verifying the connectivity and functionality of most Sun system hardware. SunVTS software can be tailored to run on various types of machines ranging from desktops to servers, and supports testing in both 32-bit and 64-bit Solaris operating environments. Tests examine subsystems such as processors, peripherals, storage, network, memory, graphics and video, audio, and communication.

Sun VTS can be used to validate a system during development or production, and for troubleshooting, periodic maintenance, and system or subsystem stressing.

Sun ONE Studio 7, Compiler Collection

Formally known as Forte™ for High Performance Computing software, the Sun™ ONE Studio 7, Compiler Collection software is a development environment for FORTRAN, C, and C++. This software includes compilers, libraries, and a debugger, which support serial, threaded, and OpenMP applications. This software is available from `http://www.sun.com`.

Sun Cluster Grid Implementation Example

FIGURE 5 is a block diagram representing an example of the implementation of a Sun Cluster Grid. Each block represents a host in one of the tiers.

At the access tier, one host is used to allow logins. This machine is configured as a SGE software submit host.

An administration host is used to access all hosts in the cluster grid through a serial interconnect. This machine also runs the console layer of SunMC software, and is configured as an SGE administration host, allowing administrative control of the SGE software.

Two hosts are also used at the management tier. One host runs the SGE master service and SunMC software. The other provides install, backup, and NFS services.

The compute layer consists of both Solaris and Linux operating environments in this example. The three SMP compute hosts can cater for OpenMP, and large memory serial jobs. A Solaris or Linux cluster interconnected by Myrinet is available for MPI jobs.
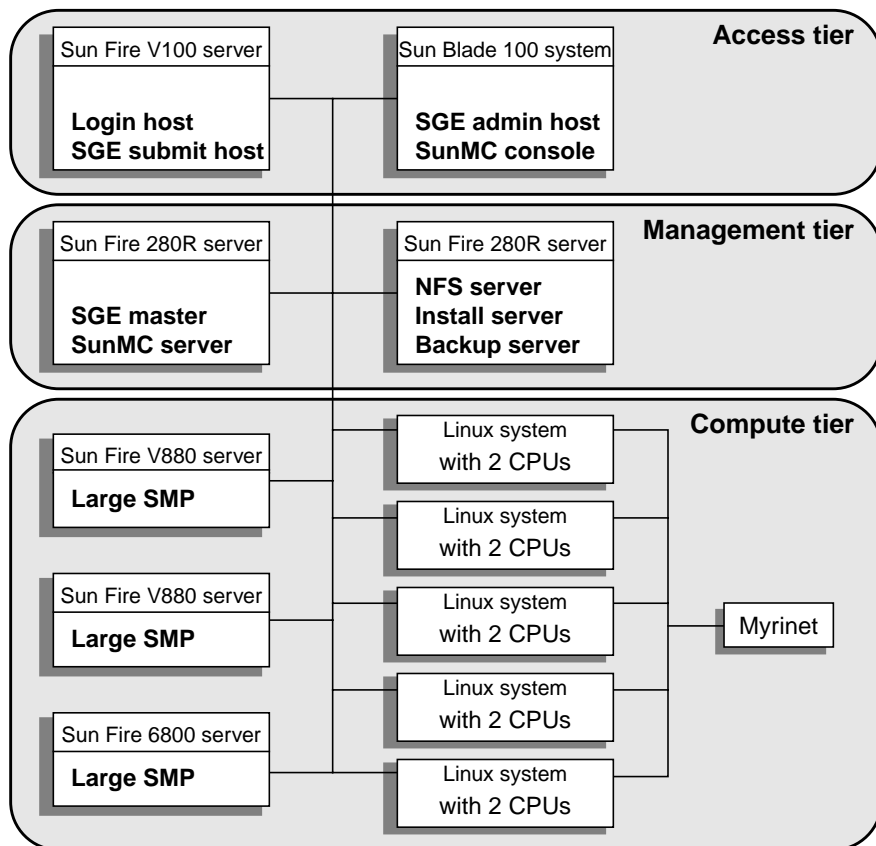
FIGURE 5   Cluster Grid Implementation Example

Of course, a vast number of alternative arrangements are valid at each tier, and full treatment of the cluster grid design and implementation will be provided in an upcoming BluePrints OnLine article entitled "Introduction to the Cluster Grid – Part 2".

About the Authors

James Coomer is a certified systems engineer in Sun's Solution Technology group, and is based in Sale, Manchester, England. He has five years experience in high performance computing, and currently works closely with UK academic institutions on high performance and grid computing. James received a M.Phys in theoretical physics at Lancaster University in 1997, and went on to gain a Ph.D. in theoretical quantum chemistry at Exeter University.

Charu Chaubal is an engineer in the Grid Computing group for Sun Microsystems. He has been working on implementations of grid technology, for customers and for demonstration projects, for the last two years. He has also developed and delivered training courses on grid computing, and performed technical marketing for grid technology products. Charu received a Bachelor of Science in Engineering from the University of Pennsylvania, and a Ph.D. from the University of California at Santa Barbara, where he studied the numerical modeling of complex fluids.

Related Resources

For additional information about the topics discussed in this article, refer to the following web sites:

For additional information about the topics discussed in this article, refer to the following web sites:

- `http://www.sun.com/software/gridware/whitepapers.html` – This site is a source for the latest Sun grid white papers.
- `http://www.sun.com/gridengine` – SGE product page. This site provides access to Sun Cluster Grid white papers, case studies, software downloads, and more.
- `http://gridengine.sunsource.net` – The open source SGE web site. This site, sponsored by Sun, and hosted by Colabnet, is a source for continued collaborative development of the Grid Engine project.
- http://www.sun.com/solutions/hpc/communitysource – Sun's HPC ClusterTools community source code access site, where Sun HPC ClusterTools source code is freely available through the Sun Community Source Code License (SCSL) model.
- http://www.sun.com/oem/products/vts/index.html – Sun's SunVTS download site where you can download the SunVTS software and documentation for free.

Ordering Sun Documents

The SunDocs℠ program provides more than 250 manuals from Sun Microsystems, Inc. If you live in the United States, Canada, Europe, or Japan, you can purchase documentation sets or individual manuals through this program.

Accessing Sun Documentation Online

The `docs.sun.com`℠ web site enables you to access Sun technical documentation online. You can browse the `docs.sun.com` archive, or search for a specific book title or subject. The URL is `http://docs.sun.com/`.

To reference Sun BluePrints OnLine articles, visit the Sun BluePrints OnLine web site at: `http://www.sun.com/blueprints/online.html`.