Sun Cluster Overview for Solaris OS

Adobe PostScript™

040203@7940

Contents

Preface

Sun™ Cluster Overview for Solaris OS introduces the Sun Cluster product by explaining the purpose of the product and the means by which Sun Cluster achieves this purpose. This book also explains key concepts for Sun Cluster. The information in this document enables you to become familiar with Sun Cluster features and functionality.

Related Documentation

Information about related Sun Cluster topics is available in the documentation that is listed in the following table. All Sun Cluster documentation is available at `http://docs.sun.com`.

Topic	Documentation
Concepts	*Sun Cluster Concepts Guide for Solaris OS*
Overview	*Sun Cluster Overview for Solaris OS*
Glossary	*Sun Java Enterprise System 2003Q4 Glossary*
Hardware administration	*Sun Cluster Hardware Administration Manual for Solaris OS*
	Individual hardware administration guides
Software installation	*Sun Cluster Software Installation Guide for Solaris OS*
Data service administration	*Sun Cluster Data Service Planning and Administration Guide for Solaris OS*
	Individual data service guides
Data service development	*Sun Cluster Data Services Developer's Guide for Solaris OS*

Topic	Documentation
System administration	*Sun Cluster System Administration Guide for Solaris OS*
Error messages	*Sun Cluster Error Messages Guide for Solaris OS*
Command and function references	*Sun Cluster Reference Manual for Solaris OS*

For a complete list of Sun Cluster documentation, see the release notes for your release of Sun Cluster software at `http://docs.sun.com`.

Accessing Sun Documentation Online

The docs.sun.com^{SM} 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`.

Ordering Sun Documentation

Sun Microsystems offers select product documentation in print. For a list of documents and how to order them, see "Buy printed documentation" at `http://docs.sun.com`.

Getting Help

If you have problems installing or using the Sun Cluster system, contact your service provider and provide the following information:

- Your name and email address (if available)
- Your company name, address, and phone number
- The model and serial numbers of your systems
- The release number of the operating environment (for example, Solaris 9)
- The release number of the Sun Cluster software (for example, 3.1 4/04)

Use the following commands to gather information about each node on your system for your service provider.

Command	Function
`prtconf -v`	Displays the size of the system memory and reports information about peripheral devices
`psrinfo -v`	Displays information about processors
`showrev -p`	Reports which patches are installed
`prtdiag -v`	Displays system diagnostic information
`scinstall -pv`	Displays Sun Cluster software release and package version information
`scstat`	Provides a snapshot of the cluster status
`scconf -p`	Lists cluster configuration information
`scrgadm -p`	Displays information about installed resources, resource groups, and resource types

Also have available the contents of the `/var/adm/messages` file.

Typographic Conventions

The following table describes the typographic changes that are used in this book.

TABLE P–1 Typographic Conventions

Typeface or Symbol	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories, and onscreen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% you have mail.`
`AaBbCc123`	What you type, contrasted with onscreen computer output	`machine_name%` **`su`** `Password:`
AaBbCc123	Command-line placeholder: replace with a real name or value	To delete a file, type **`rm`** *filename*.
AaBbCc123	Book titles, new terms, or terms to be emphasized	Read Chapter 6 in *User's Guide*. These are called *class* options. You must be *root* to do this.

Shell Prompts in Command Examples

The following table shows the default system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

TABLE P–2 Shell Prompts

Shell	Prompt
C shell prompt	`machine_name%`
C shell superuser prompt	`machine_name#`
Bourne shell and Korn shell prompt	`$`
Bourne shell and Korn shell superuser prompt	`#`

Introduction to Sun Cluster

The SunPlex system is an integrated hardware and Sun Cluster software solution that is used to create highly available and scalable services. This chapter provides a high-level overview of Sun Cluster features.

This chapter contains the following sections:

- "Making Applications Highly Available With Sun Cluster" on page 9
- "Monitoring Failure" on page 13
- "Administration and Configuration Tools" on page 13

Making Applications Highly Available With Sun Cluster

A cluster is two or more systems, or nodes, that work together as a single, continuously available system to provide applications, system resources, and data to users. Each node on a cluster is a fully functional standalone system. However, in a clustered environment, the nodes are connected by an interconnect and work together as a single entity to provide increased availability and performance.

Highly available clusters provide nearly continuous access to data and applications by keeping the cluster running through failures that would normally bring down a single server system. No single failure—hardware, software, or network—can cause a cluster to fail. By contrast, fault-tolerant hardware systems provide constant access to data and applications, but at a higher cost because of specialized hardware. Fault-tolerant systems usually have no provision for software failures.

Each Sun Cluster system is a collection of tightly coupled nodes that provide a single administration view of network services and applications. The Sun Cluster system achieves high availability through a combination of the following hardware and software:

- Redundant disk systems provide storage. These disk systems are generally mirrored to permit uninterrupted operation if a disk or subsystem fails. Redundant connections to the disk systems ensures that data is not isolated if a server, controller, or cable fails. A high-speed interconnect among nodes provides access to resources. All nodes in the cluster are also connected to a public network, enabling clients on multiple networks to access the cluster.

- Redundant hot-swappable components, such as power supplies and cooling systems, improve availability by enabling systems to continue operation after a hardware failure. Hot-swappable components provide the ability to add or remove hardware components in a functioning system without bringing it down.

- Sun Cluster software's high-availability framework detects a node failure quickly and migrates the application or service to another node that runs in an identical environment. At no time are all applications unavailable. Applications unaffected by a down node are fully available during recovery. Furthermore, applications of the failed node become available as soon as they are recovered. A recovered application does not have to wait for all other applications to complete their recovery.

Availability Management

An application is highly available if it survives any single software or hardware failure in the system. Failures that are caused by bugs or data corruption within the application itself are excluded. The following apply to highly available applications:

- Recovery is transparent from the applications that use a resource.
- Resource access is fully preserved across node failure.
- Applications cannot detect that the hosting node has been moved to another node.
- Failure of a single node is completely transparent to programs on remaining nodes that use the files, devices, and disk volumes attached to this node.

Failover and Scalable Services and Parallel Applications

Failover and scalable services and parallel applications enable you to make your applications highly available and to improve an application's performance on a cluster.

A failover service provides high availability through redundancy. When a failure occurs, you can configure an application that is running to either restart on the same node, or be moved to another node in the cluster, without user intervention.

To increase performance, a scalable service leverages the multiple nodes in a cluster to concurrently run an application. In a scalable configuration, each node in the cluster can provide data and process client requests.

Parallel databases enable multiple instances of the database server to do the following:

- Participate in the cluster
- Handle different queries on the same database simultaneously
- Provide parallel query capability on large queries

For more information about failover and scalable services and parallel applications, see "Data Service Types" on page 26.

IP Network Multipathing

Clients make data requests to the cluster through the public network. Each cluster node is connected to at least one public network through one or multiple public network adapters.

IP network multipathing enables a server to have multiple network ports connected to the same subnet. First, IP network multipathing software provides resilience from network adapter failure by detecting the failure or repair of a network adapter. The software then simultaneously switches the network address to and from the alternative adapter. When more than one network adapter is functional, IP network multipathing increases data throughput by spreading outbound packets across adapters.

Storage Management

Multihost storage makes disks highly available by connecting the disks to multiple nodes. Multiple nodes enable multiple paths to access the data, if one path fails, another one is available to take its place.

Multihost disks enable the following cluster processes:

- Tolerating single-node failures.
- Centralizing application data, application binaries, and configuration files.
- Protecting against node failures. If client requests are accessing the data through a node that fails, the requests are switched over to use another node that has a direct connection to the same disks.
- Providing access either globally through a primary node that "masters" the disks, or by direct concurrent access through local paths.

Volume Management Support

A volume manager enables you to manage large numbers of disks and the data on those disks. Volume managers can increase storage capacity and data availability by offering the following features:

- Disk-drive striping and concatenation
- Disk-mirroring
- Disk-drive hot spares
- Disk-failure handling and disk replacements

Sun Cluster systems support the following volume managers:

- Solaris Volume Manager
- VERITAS Volume Manager

Sun StorEdge Traffic Manager

Sun StorEdge Traffic Manager software is fully integrated starting with the Solaris Operating System 8 core I/O framework. Sun StorEdge Traffic Manager software enables you more effectively to represent and manage devices that are accessible through multiple I/O controller interfaces within a single instance of the Solaris operating environment. The Sun StorEdge Traffic Manager architecture enables the following:

- Protection against I/O outages due to I/O controller failures
- Automatic switches to an alternate controller upon an I/O controller failure
- Increased I/O performance by load balancing across multiple I/O channels

Hardware Redundant Array of Independent Disks Support

Sun Cluster systems support the use of hardware Redundant Array of Independent Disks (RAID) and host-based software RAID. Hardware RAID uses the storage array's or storage system's hardware redundancy to ensure that independent hardware failures do not impact data availability. If you mirror across separate storage arrays, host-based software RAID ensures that independent hardware failures do not impact data availability when an entire storage array is offline. Although you can use hardware RAID and host-based software RAID concurrently, you need only one RAID solution to maintain a high degree of data availability.

File System Support

Because one of the inherent properties of clustered systems is shared resources, a cluster requires a file system that addresses the need for files to be shared coherently. The Sun Cluster file system enables users or applications to access any file on any

node of the cluster by using the remote or local standard UNIX APIs. If an application is moved from one node to another node, no change is required for the application to access the same files. No changes need to be made to existing applications to fully utilize the cluster file system.

Monitoring Failure

The Sun Cluster system makes the path between users and data highly available by using multihost disks, multipathing, and a global file system. The Sun Cluster system monitors failures for the following:

- Applications – Most of the Sun Cluster data services supply a fault monitor that periodically probes the data service to determine its health. A fault monitor verifies that the application daemon or daemons are running and that clients are being served. Based on the information that is returned by probes, a predefined action such as restarting daemons or causing a failover can be initiated.
- Disk-Paths – Sun Cluster software supports disk-path monitoring (DPM). DPM improves the overall reliability of failover and switchover by reporting the failure of a secondary disk path.
- Internet Protocol (IP) Multipath – Solaris IP network multipathing software on Sun Cluster systems provide the basic mechanism for monitoring public network adapters. IP multipathing also enables failover of IP addresses from one adapter to another adapter when a fault is detected.

Administration and Configuration Tools

You can install, configure, and administer the Sun Cluster system either though the SunPlex Manager GUI or through the command-line interface (CLI).

The Sun Cluster system also has a module that runs as part of Sun Management Center software that provides a GUI to certain cluster tasks.

SunPlex Manager

SunPlex Manager is a browser-based tool for administering Sun Cluster systems. The SunPlex Manager software enables administrators to perform system management and monitoring, software installation, and system configuration.

The SunPlex Manager software includes the following features.

- Built-in security and authorization mechanisms
- Secure Sockets Layer (SSL) support
- Role-based access control (RBAC)
- Pluggable Authentication Module (PAM)
- NAFO and IP network multipathing group administration facilities
- Quorum devices, transports, shared storage device, and resource group administration
- Sophisticated error checking and autodetection of private interconnects

Command-Line Interface

The Sun Cluster command-line interface is a set of utilities you can use to install and administer Sun Cluster systems, and administer the volume manager portion of Sun Cluster software.

You can perform the following SunPlex administration tasks through the Sun Cluster CLI:

- Validating a Sun Cluster configuration
- Installing and configuring Sun Cluster software
- Updating a Sun Cluster configuration
- Managing the registration of resource types, the creation of resource groups, and the activation of resources within a resource group
- Changing node mastery and states for resource groups and disk device groups
- Controlling access with role-based access control (RBAC)
- Shutting down the entire cluster

Sun Management Center

The Sun Cluster system also has a module that runs as part of Sun Management Center software. Sun Management Center software serves as the cluster's base for administrative and monitoring operations and enables system administrators to perform the following tasks through a GUI or CLI:

- Configuring a remote system
- Monitoring performance
- Dectecting and isolating hardware and software faults

Sun Management Center software can also be used as the interface to manage dynamic reconfiguration within Sun Cluster servers. Dynamic reconfiguration includes domain creation, dynamic board attach, and dynamic detach.

Role-Based Access Control

In conventional UNIX systems, the root user, also referred to as superuser, is omnipotent, with the ability to read and write to any file, run all programs, and send kill signals to any process. Solaris role-based access control (RBAC) is an alternative to the all-or-nothing superuser model. RBAC uses the security principle of least privilege, which is that no user should be given more privilege than necessary for performing his or her job.

RBAC enables an organization to separate superuser capabilities and package them into special user accounts or roles for assignment to specific individuals. This separation and packaging enables a variety of security policies. Accounts can be set up for special-purpose administrators in such areas as security, networking, firewall, backups, and system operation.

Key Concepts for Sun Cluster

This chapter explains the key concepts related to the hardware and software components of the Sun Cluster system that you need to understand before working with Sun Cluster systems.

This chapter contains the following sections:

- "Cluster Nodes" on page 17
- "Cluster Interconnect" on page 18
- "Cluster Membership" on page 18
- "Cluster Configuration Repository" on page 19
- "Fault Monitors" on page 19
- "Quorum Devices" on page 20
- "Devices" on page 23
- "Data Services" on page 24

Cluster Nodes

A cluster node is a machine that runs both the Solaris software and Sun Cluster software. The Sun Cluster software enables you to have from two to eight nodes in a cluster.

Cluster nodes are generally attached to one or more disks. Nodes not attached to disks use the cluster file system to access the multihost disks. Nodes in parallel database configurations share concurrent access to some or all disks.

Every node in the cluster is aware when another node joins or leaves the cluster. Also, every node in the cluster is aware of the resources that are running locally as well as the resources that are running on the other cluster nodes.

Nodes in the same cluster should have similar processing, memory, and I/O capability to enable failover to occur without significant degradation in performance. Because of the possibility of failover, each node should have sufficient capacity to meet service level agreements if a node fails.

Cluster Interconnect

The cluster interconnect is the physical configuration of devices that are used to transfer cluster-private communications and data service communications between cluster nodes.

Redundant interconnects enable operation to continue over the surviving interconnects while system administrators isolate failures and repair communication. The Sun Cluster software detects, repairs, and automatically reinitiates communication over a repaired interconnect.

For more information, see "Cluster Interconnect" on page 35.

Cluster Membership

The Cluster Membership Monitor (CMM) is a distributed set of agents that exchange messages over the cluster interconnect to complete the following tasks:

- Enforcing a consistent membership view on all nodes (quorum)
- Driving synchronized reconfiguration in response to membership changes
- Handling cluster partitioning
- Ensuring full connectivity among all cluster members by leaving unhealthy nodes out of the cluster until it is repaired

The main function of the CMM is to establish cluster membership, which requires a cluster-wide agreement on the set of nodes that participate in the cluster at any time. The CMM detects major cluster status changes on each node, such as loss of communication between one or more nodes. The CMM relies on the transport kernel module to generate heartbeats across the transport medium to other nodes in the cluster. When the CMM does not detect a heartbeat from a node within a defined time-out period, the CMM considers the node to have failed and the CMM initiates a cluster reconfiguration to renegotiate cluster membership.

To determine cluster membership and to ensure data integrity, the CMM performs the following tasks:

- Accounting for a change in cluster membership, such as a node joining or leaving the cluster
- Ensuring that an unhealthy node leaves the cluster
- Ensuring that an unhealthy node remains inactive until it is repaired
- Preventing the cluster from partitioning itself into subsets of nodes

See "Data Integrity" on page 21 for more information about how the cluster protects itself from partitioning into multiple separate clusters.

Cluster Configuration Repository

The Cluster Configuration Repository (CCR) is a private, cluster-wide, distributed database for storing information that pertains to the configuration and state of the cluster. To avoid corrupting configuration data, each node must be aware of the current state of the cluster resources. The CCR ensures that all nodes have a consistent view of the cluster. The CCR is updated when error or recovery situations occur or when the general status of the cluster changes.

The CCR structures contain the following types of information:

- Cluster and node names
- Cluster transport configuration
- The names of Solaris Volume Manager disk sets or VERITAS disk groups
- A list of nodes that can master each disk group
- Operational parameter values for data services
- Paths to data service callback methods
- DID device configuration
- Current cluster status

Fault Monitors

Sun Cluster system makes all components on the "path" between users and data highly available by monitoring the applications themselves, the file system, and network interfaces.

The Sun Cluster software detects a node failure quickly and creates an equivalent server for the resources on the failed node. The Sun Cluster software ensures that resources unaffected by the failed node are constantly available during the recovery and that resources of the failed node become available as soon as they are recovered.

Data Services Monitoring

Each Sun Cluster data service supplies a fault monitor that periodically probes the data service to determine its health. A fault monitor verifies that the application daemon or daemons are running and that clients are being served. Based on the information returned by probes, predefined actions such as restarting daemons or causing a failover, can be initiated.

Disk-Path Monitoring

Sun Cluster software supports disk-path monitoring (DPM). DPM improves the overall reliability of failover and switchover by reporting the failure of a secondary disk-path. You can use one of two methods for monitoring disk paths. The first method is provided by the scdpm command. This command enables you to monitor, unmonitor, or display the status of disk paths in your cluster. See the scdpm(1M) man page for more information about command-line options.

The second method for monitoring disk paths in your cluster is provided by the SunPlex Manager graphical user interface (GUI). SunPlex Manager provides a topological view of the monitored disk paths. The view is updated every 10 minutes to provide information about the number of failed pings.

IP Multipath Monitoring

Each cluster node has its own IP network multipathing configuration, which can differ from the configuration on other cluster nodes. IP network multipathing monitors the following network communication failures:

- The transmit and receive path of the network adapter has stopped transmitting packets.
- The attachment of the network adapter to the link is down.
- The port on the switch does not transmit-receive packets.
- The physical interface in a group is not present at system boot.

Quorum Devices

A quorum device is a disk shared by two or more nodes that contributes votes that are used to establish a quorum for the cluster to run. The cluster can operate only when a quorum of votes is available. The quorum device is used when a cluster becomes partitioned into separate sets of nodes to establish which set of nodes constitutes the new cluster.

Both cluster nodes and quorum devices vote to form quorum. By default, cluster nodes acquire a quorum vote count of one when they boot and become cluster members. Nodes can have a vote count of zero when the node is being installed, or when an administrator has placed a node into the maintenance state.

Quorum devices acquire quorum vote counts that are based on the number of node connections to the device. When you set up a quorum device, it acquires a maximum vote count of N-1 where N is the number of connected votes to the quorum device. For example, a quorum device that is connected to two nodes with nonzero vote counts has a quorum count of one (two minus one).

Data Integrity

The Sun Cluster system attempts to prevent data corruption and ensure data integrity. Because cluster nodes share data and resources, a cluster must never split into separate partitions that are active at the same time. The CMM guarantees that only one cluster is operational at any time.

Two types of problems can arise from cluster partitions: split brain and amnesia. Split brain occurs when the cluster interconnect between nodes is lost and the cluster becomes partitioned into subclusters, and each subcluster believes that it is the only partition. A subcluster that is not aware of the other subclusters could cause a conflict in shared resources such as duplicate network addresses and data corruption.

Amnesia occurs if all the nodes leave the cluster in staggered groups. An example is a two-node cluster with nodes A and B. If node A goes down, the configuration data in the CCR is updated on node B only, and not node A. If node B goes down at a later time, and if node A is rebooted, node A will be running with old contents of the CCR. This state is called amnesia and might lead to running a cluster with stale configuration information.

You can avoid split brain and amnesia by giving each node one vote and mandating a majority of votes for an operational cluster. A partition with the majority of votes has a quorum and is enabled to operate. This majority vote mechanism works well if more than two nodes are in the cluster. In a two-node cluster, a majority is two. If such a cluster becomes partitioned, an external vote enables a partition to gain quorum. This external vote is provided by a quorum device. A quorum device can be any disk that is shared between the two nodes.

Table 2–1 describes how Sun Cluster software uses quorum to avoid split brain and amnesia.

TABLE 2–1 Cluster Quorum, and Split-Brain and Amnesia Problems

Partition Type	Quorum Solution
Split brain	Enables only the partition (subcluster) with a majority of votes to run as the cluster (only one partition can exist with such a majority). After a node loses the race for quorum, that node panics.
Amnesia	Guarantees that when a cluster is booted, it has at least one node that was a member of the most recent cluster membership (and thus has the latest configuration data).

Failure Fencing

A major issue for clusters is a failure that causes the cluster to become partitioned (called *split brain*). When this situation occurs, not all nodes can communicate, so individual nodes or subsets of nodes might try to form individual or subset clusters. Each subset or partition might "believe" it has sole access and ownership to the multihost disks. Attempts by multiple nodes to write to the disks can result in data corruption.

Failure fencing limits node access to multihost disks by preventing access to the disks. When a node leaves the cluster (it either fails or becomes partitioned), failure fencing ensures that the node can no longer access the disks. Only current member nodes have access to the disks, ensuring data integrity.

The Sun Cluster system uses SCSI disk reservations to implement failure fencing. Using SCSI reservations, failed nodes are "fenced" away from the multihost disks, preventing them from accessing those disks.

When a cluster member detects that another node is no longer communicating over the cluster interconnect, it initiates a failure-fencing procedure to prevent the failed node from accessing shared disks. When this failure fencing occurs, the fenced node panics and a "reservation conflict" message is displayed on its console.

Failfast Mechanism for Failure Fencing

The failfast mechanism panics a failed node, but it does not prevent the failed node from rebooting. After the panic, the node might reboot and attempt to rejoin the cluster.

If a node loses connectivity to other nodes in the cluster, and it is not part of a partition that can achieve quorum, it is forcibly removed from the cluster by another node. Another node that is part of the partition that can achieve quorum places reservations on the shared disks. The node that does not have quorum then panics as a result of the failfast mechanism.

Devices

The global file system makes all files across a cluster equally accessible and visible to all nodes. Similarly, Sun Cluster software makes all devices on a cluster accessible and visible throughout the cluster. That is, the I/O subsystem enables access to any device in the cluster, from any node, without regard to where the device is physically attached. This access is referred to as global device access.

Global Devices

Sun Cluster systems use global devices to provide cluster-wide, highly available access to any device in a cluster, from any node. Generally, if a node fails while providing access to a global device, the Sun Cluster software switches over to another path to the device and redirects the access to that path. This redirection is easy with global devices because the same name is used for the device regardless of the path. Access to a remote device is performed in the same way as on a local device that uses the same name. Also, the API to access a global device on a cluster is the same as the API that is used to access a device locally.

Sun Cluster global devices include disks, CD-ROMs, and tapes. However, disks are the only multiported global devices that are supported. This limited support means that CD-ROM and tape devices are not currently highly available devices. The local disks on each server are also not multiported, and thus are not highly available devices.

The cluster assigns unique IDs to each disk, CD-ROM, and tape device in the cluster. This assignment enables consistent access to each device from any node in the cluster.

Device ID

The Sun Cluster software manages global devices through a construct that is known as the device ID (DID) driver. This driver is used to automatically assign unique IDs to every device in the cluster, including multihost disks, tape drives, and CD-ROMs.

The DID driver is an integral part of the global device access feature of the cluster. The DID driver probes all nodes of the cluster and builds a list of unique disk devices. The DID driver also assigns each device a unique major and minor number that is consistent on all nodes of the cluster. Access to the global devices is through the unique DID assigned by the DID driver instead of the traditional Solaris DIDs.

This approach ensures that any application accessing disks, such as Solaris Volume Manager or Sun Java System Directory Server, uses a consistent path across the cluster. This consistency is especially important for multihost disks, because the local major and minor numbers for each device can vary from node to node. These numbers can change the Solaris device naming conventions as well.

Local Devices

The Sun Cluster software also manages local devices. These devices are accessible only on a node that is running a service and has a physical connection to the cluster. Local devices can have a performance benefit over global devices because local devices do not have to replicate state information on multiple nodes simultaneously. The failure of the domain of the device removes access to the device unless the device can be shared by multiple nodes.

Disk Device Groups

Disk device groups enable volume manager disk groups to become "global" because it provides multipath and multihost support to the underlying disks. Each cluster node physically attached to the multihost disks provides a path to the disk device group.

In the Sun Cluster system, multihost disks can be under control of the Sun Cluster software by being registering as disk device groups. This registration provides the Sun Cluster system with information about which nodes have a path to what volume manger disk groups. The Sun Cluster software creates a raw disk device group for each disk and tape device in the cluster. These cluster device groups remain in an offline state until you access them as global devices either by mounting a global file system or by accessing a raw database file.

Data Services

A data service is the combination of software and configuration files that enables an application to run without modification in a Sun Cluster configuration. When running in a Sun Cluster configuration, an application runs as a resource under the control of the Resource Group Manager (RGM). A data service enables you to configure an application such as Sun Java System Web Server or Oracle database to run on a cluster instead of on a single server.

The software of a data service provides implementations of Sun Cluster management methods that perform the following operations on the application:

- Starting the application
- Stopping the application
- Monitoring faults in the application and recovering from these faults

The configuration files of a data service define the properties of the resource that represents the application to the RGM.

The RGM controls the disposition of the failover and scalable data services in the cluster. The RGM is responsible for starting and stopping the data services on selected nodes of the cluster in response to cluster membership changes. The RGM enables data service applications to utilize the cluster framework.

The RGM controls data services as resources. These implementations are either supplied by Sun or created by a developer who uses a generic data service template, the Data Service Development Library API (DSDL API), or the Resource Management API (RMAPI). The cluster administrator creates and manages resources in containers that are called resource groups. RGM and administrator actions cause resources and resource groups to move between online and offline states.

Resource Types

A resource type is a collection of properties that describe an application to the cluster. This collection includes information about how the application is to be started, stopped, and monitored on nodes of the cluster. A resource type also includes application-specific properties that need to be defined in order to use the application in the cluster. Sun Cluster data services has several predefined resource types. For example, Sun Cluster HA for Oracle is the resource type `SUNW.oracle-server` and Sun Cluster HA for Apache is the resource type `SUNW.apache`.

Resources

A resource is an instance of a resource type that is defined cluster wide. The resource type enables multiple instances of an application to be installed on the cluster. When you initialize a resource, the RGM assigns values to application-specific properties and the resource inherits any properties on the resource type level.

Data services utilize several types of resources. Applications such as Apache Web Server or Sun Java System Web Server utilize network addresses (logical hostnames and shared addresses) on which the applications depend. Application and network resources form a basic unit that is managed by the RGM.

Resource Groups

Resources that are managed by the RGM are placed into resource groups so that they can be managed as a unit. A resource group is a set of related or interdependent resources. For example, a resource derived from a `SUNW.LogicalHostname` resource type might be placed in the same resource group as a resource derived from an Oracle database resource type. A resource group migrates as a unit if a failover or switchover is initiated on the resource group.

Data Service Types

Data services enable applications to become highly available and scalable services help prevent significant application interruption after any single failure within the cluster.

When you configure a data service, you must configure the data service as one of the following data service types:

- Failover data service
- Scalable data service
- Parallel data service

Failover Data Services

Failover is the process by which the cluster automatically relocates an application from a failed primary node to a designated redundant secondary node. Failover applications have the following characteristics:

- Capable of running on only one node of the cluster
- Not cluster-aware
- Dependent on the cluster framework for high availability

If the fault monitor detects an error, it either attempts to restart the instance on the same node, or to start the instance on another node (failover), depending on how the data service has been configured. Failover services use a failover resource group, which is a container for application instance resources and network resources (logical hostnames). Logical hostnames are IP addresses that can be configured up on one node, and later, automatically configured down on the original node and configured up on another node.

Clients might have a brief interruption in service and might need to reconnect after the failover has finished. However, clients are not aware of the change in the physical server that is providing the service.

Scalable Data Services

The scalable data service enables application instances to run on multiple nodes simultaneously. Scalable services use two resource groups. The scalable resource group contains the application resources and the failover resource group contains the network resources (shared addresses) on which the scalable service depends. The scalable resource group can be online on multiple nodes, so multiple instances of the service can be running simultaneously. The failover resource group that hosts the shared address is online on only one node at a time. All nodes that host a scalable service use the same shared address to host the service.

The cluster receives service requests through a single network interface (the global interface). These requests are distributed to the nodes, based on one of several predefined algorithms that are set by the load-balancing policy. The cluster can use the load-balancing policy to balance the service load between several nodes.

Parallel Applications

Sun Cluster systems provide an environment that shares parallel execution of applications across all the nodes of the cluster by using parallel databases. Sun Cluster Support for Oracle Parallel Server/Real Application Clusters is a set of packages that, when installed, enables Oracle Parallel Server/Real Application Clusters to run on Sun Cluster nodes. This data service also enables Sun Cluster Support for Oracle Parallel Server/Real Application Clusters to be managed by using Sun Cluster commands.

A parallel application has been instrumented to run in a cluster environment so that the application can be mastered by two or more nodes simultaneously. In an Oracle Parallel Server/Real Application Clusters environment, multiple Oracle instances cooperate to provide access to the same shared database. The Oracle clients can use any of the instances to access the database. Thus, if one or more instances have failed, clients can connect to a surviving instance and continue to access the database.

Sun Cluster Architecture

Sun Cluster architecture permits a group of systems to be deployed, managed, and viewed as a single, large system.

This chapter contains the following sections:

- "Sun Cluster Hardware Environment" on page 29
- "Sun Cluster Software Environment" on page 30
- "Scalable Data Services" on page 33
- "Multihost Disk Storage" on page 35
- "Cluster Interconnect" on page 35
- "IP Network Multipathing Groups" on page 36

Sun Cluster Hardware Environment

The following hardware components make up a cluster:

- Cluster nodes with local disks (unshared) provide the main computing platform of the cluster.

- Multihost storage provides disks that are shared between nodes.

- Removable media are configured as global devices, such as tapes and CD-ROM.

- Cluster interconnect provides a channel for internode communication.

- Public network interfaces enable the network interfaces that are used by client systems to access data services on the cluster.

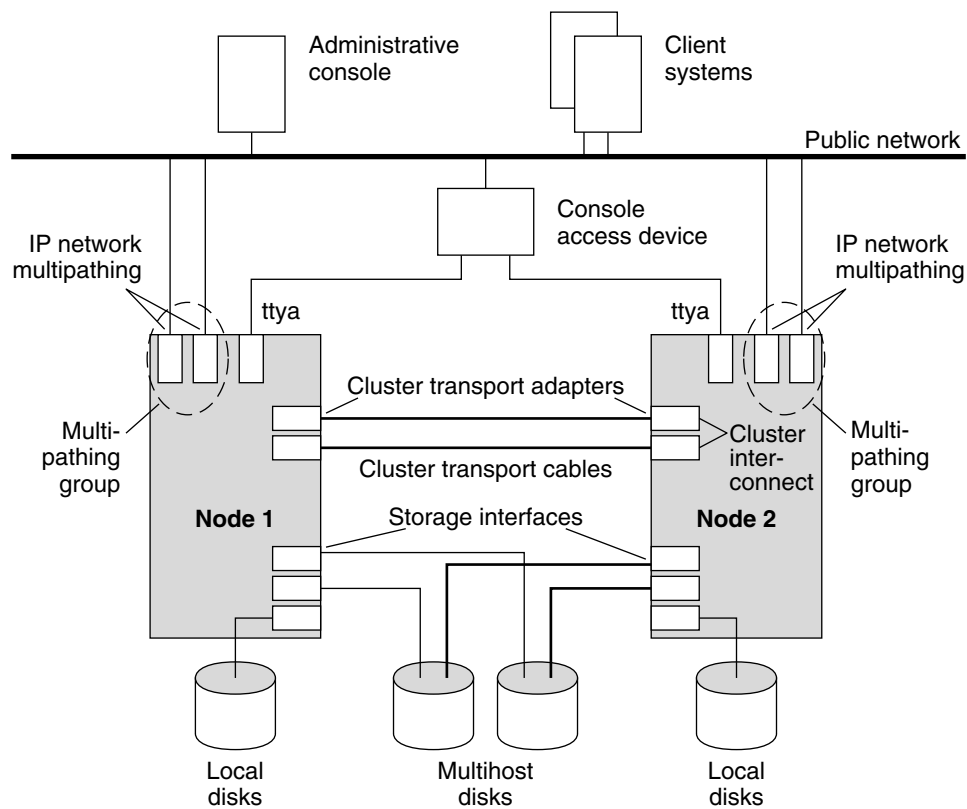Figure 3–1 illustrates how the hardware components work with each other.

FIGURE 3–1 Sun Cluster Hardware Components

Sun Cluster Software Environment

To function as a cluster member, a node must have the following software installed:

- Solaris software
- Sun Cluster software
- Data service application
- Volume management (Solaris™ Volume Manager or VERITAS Volume Manager)

  An exception is a configuration that uses volume management on the box. This configuration might not require a software volume manager.

Figure 3–2 shows a high-level view of the software components that work together to create the Sun Cluster software environment.
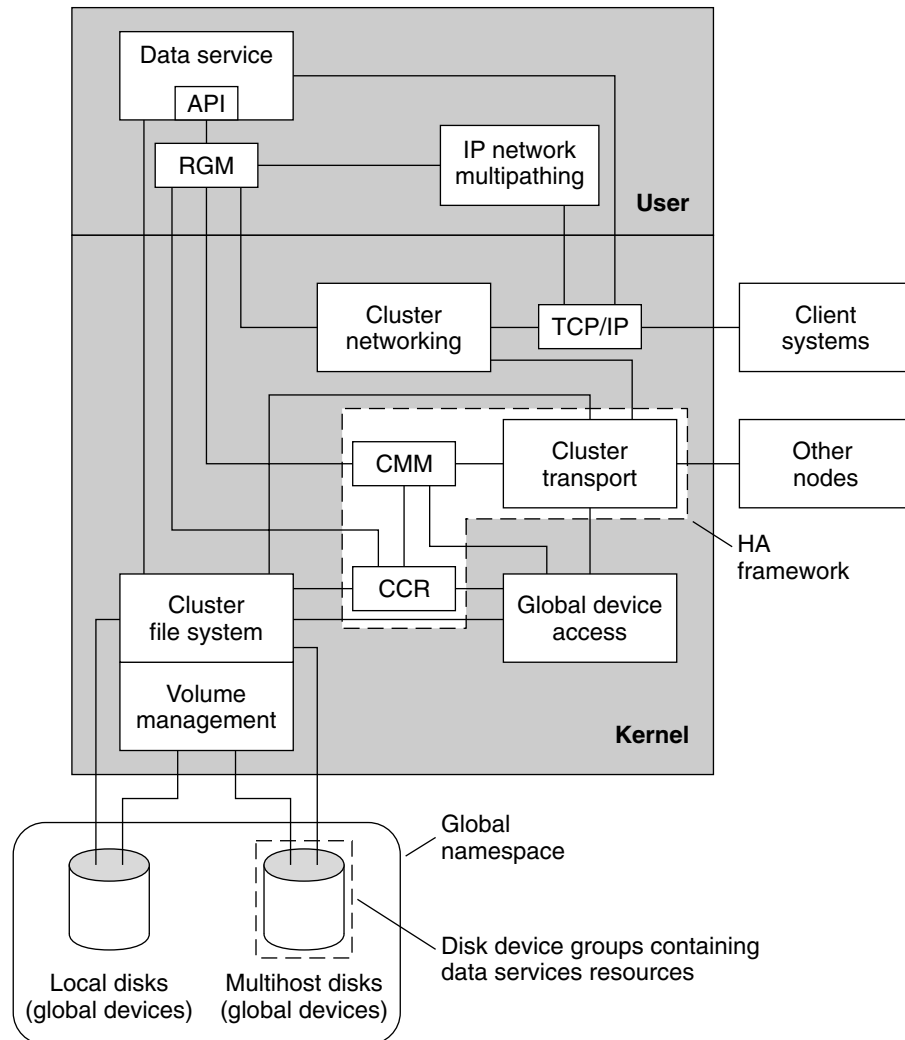
FIGURE 3–2 Sun Cluster Software Architecture

Cluster Membership Monitor

To ensure that data is safe from corruption, all nodes must reach a consistent agreement on the cluster membership. When necessary, the CMM coordinates a cluster reconfiguration of cluster services in response to a failure.

The CMM receives information about connectivity to other nodes from the cluster transport layer. The CMM uses the cluster interconnect to exchange state information during a reconfiguration.

After detecting a change in cluster membership, the CMM performs a synchronized configuration of the cluster. In this configuration, cluster resources might be redistributed, based on the new membership of the cluster.

The CMM runs entirely in the kernel.

Cluster Configuration Repository (CCR)

The CCR relies on the CMM to guarantee that a cluster is running only when quorum is established. The CCR is responsible for verifying data consistency across the cluster, performing recovery as necessary, and facilitating updates to the data.

Cluster File Systems

A cluster file system is a proxy between the following:

- The kernel on one node and the underlying file system
- The volume manager running on a node that has a physical connection to the disk or disks

Cluster file systems are dependent on global devices (disks, tapes, CD-ROMs). The global devices can be accessed from any node in the cluster through the same file name (for example, /dev/global/). That node does not need a physical connection to the storage device. You can use a global device the same as a regular device, that is, you can create a file system on a global device by using newfs or mkfs.

The cluster file system has the following features:

- File access locations are transparent. A process can open a file that is located anywhere in the system. Also, processes on all nodes can use the same path name to locate a file.

  Note – When the cluster file system reads files, it does not update the access time on those files.

- Coherency protocols are used to preserve the UNIX file access semantics even if the file is accessed concurrently from multiple nodes.
- Extensive caching is used with zero-copy bulk I/O movement to move file data efficiently.

- The cluster file system provides highly available advisory file-locking functionality by using the `fcntl(2)` interfaces. Applications that run on multiple cluster nodes can synchronize access to data by using advisory file locking on a cluster file system file. File locks are recovered immediately from nodes that leave the cluster, and from applications that fail while holding locks.

- Continuous access to data is ensured, even when failures occur. Applications are not affected by failures if a path to disks is still operational. This guarantee is maintained for raw disk access and all file system operations.

- Cluster file systems are independent from the underlying file system and volume management software. Cluster file systems make any supported on-disk file system global.

Scalable Data Services

The primary goal of cluster networking is to provide scalability for data services. Scalability means that as the load offered to a service increases, a data service can maintain a constant response time to this increased workload as new nodes are added to the cluster and new server instances are run. A good example of a scalable data service is a web service. Typically, a scalable data service is composed of several instances, each of which runs on different nodes of the cluster. Together, these instances behave as a single service for a remote client of that service and implement the functionality of the service. A scalable web service with several `httpd` daemons that run on different nodes can have any daemon serve a client request. The daemon that serves the request depends on a *load-balancing policy*. The reply to the client appears to come from the service, not the particular daemon that serviced the request, thus preserving the single-service appearance.

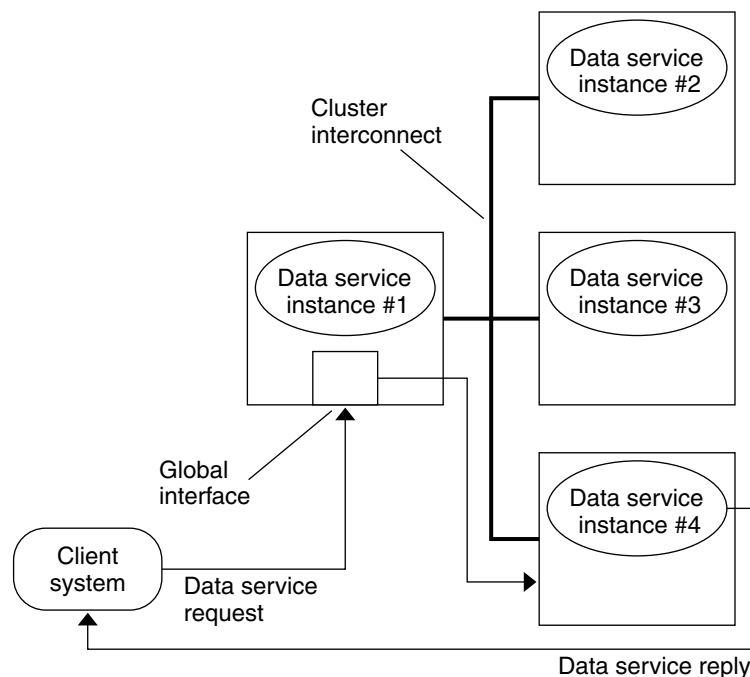The following figure depicts the scalable service architecture.

FIGURE 3–3 Scalable Data Service Architecture

The nodes that are not hosting the global interface (proxy nodes) have the shared address hosted on their loopback interfaces. Packets that are coming into the global interface are distributed to other cluster nodes, based on configurable load-balancing policies. The possible load-balancing policies are described next.

Load-Balancing Policies

Load balancing improves performance of the scalable service, both in response time and in throughput.

Two classes of scalable data services exist: *pure* and *sticky*. A pure service is one where any instance can respond to client requests. A sticky service has the cluster balancing the load for requests to the node. Those requests are not redirected to other instances.

A pure service uses a weighted load-balancing policy. Under this load-balancing policy, client requests are by default uniformly distributed over the server instances in the cluster. For example, in a three-node cluster where each node has the weight of 1, each node services one-third of the requests from any client on behalf of that service. Weights can be changed at any time through the scrgadm(1M) command interface or through the SunPlex Manager GUI.

A sticky service has two types: *ordinary sticky* and *wildcard sticky*. Sticky services allow concurrent application-level sessions over multiple TCP connections to share in-state memory (application session state).

Ordinary sticky services permit a client to share state between multiple concurrent TCP connections. The client is said to be "sticky" toward the server instance listening on a single port. The client is guaranteed that all requests go to the same server instance, if that instance remains up and accessible and the load balancing policy is not changed while the service is online.

Wildcard sticky services use dynamically assigned port numbers, but still expect client requests to go to the same node. The client is "sticky wildcard" over ports toward the same IP address.

Multihost Disk Storage

Sun Cluster software makes disks highly available by utilizing multihost disk storage, which can be connected to more than one node at a time. Volume management software can be used to arrange these disks into shared storage that is mastered by a cluster node. The disks are then configured to move to another node if a failure occurs. The use of multihosted disks in Sun Cluster systems provides a variety of benefits, including the following:

- Global access to file systems
- Multiple access paths to file systems and data
- Tolerance for single-node failures

Cluster Interconnect

All nodes must be connected by the cluster interconnect through at least two redundant physically independent networks, or paths, to avoid a single point of failure. While two interconnects are required for redundancy, up to six can be used to spread traffic to avoid bottlenecks and improve redundancy and scalability. The Sun Cluster interconnect uses Fast Ethernet, Gigabit-Ethernet, Sun Fire Link, or the Scalable Coherent Interface (SCI, IEEE 1596-1992), enabling high-performance cluster-private communications.

In clustered environments, high-speed, low-latency interconnects and protocols for internode communications are essential. The SCI interconnect in Sun Cluster systems offers improved performance over typical network interface cards (NICs). Sun Cluster uses the Remote Shared Memory (RSM™) interface for internode communication across a Sun Fire Link network. RSM is a Sun messaging interface that is highly efficient for remote memory operations.

The cluster interconnect consists of the following hardware components:

- *Adapters* – The network interface cards that reside in each cluster node. A network adapter with multiple interfaces could become a single point of failure if the entire adapter fails.

- *Junctions* – The switches that reside outside of the cluster nodes. Junctions perform pass-through and switching functions to enable you to connect more than two nodes. In a two-node cluster, you do not need junctions because the nodes can be directly connected to each other through redundant physical cables. Those redundant cables are connected to redundant adapters on each node. Greater than two-node configurations require junctions.

- *Cables* – The physical connections that are placed between either two network adapters or an adapter and a junction.

Figure 3–4 shows how the three components are connected.
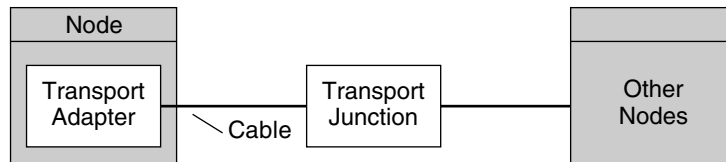
FIGURE 3–4 Cluster Interconnect

IP Network Multipathing Groups

Public network adapters are organized into IP multipathing groups (multipathing groups). Each multipathing group has one or more public network adapters. Each adapter in a multipathing group can be active, or you can configure standby interfaces that are inactive unless a failover occurs.

Multipathing groups provide the foundation for logical hostname and shared address resources. The same multipathing group on a node can host any number of logical hostname or shared address resources. To monitor public network connectivity of cluster nodes, you can create multipathing.

For more information about logical hostname and shared address resources, see the *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

Public Network Interfaces

Clients connect to the cluster through the public network interfaces. Each network adapter card can connect to one or more public networks, depending on whether the card has multiple hardware interfaces. You can set up nodes to include multiple public network interface cards that are configured so that multiple cards are active, and serve as failover backups for one another. If one of the adapters fails, the Solaris Internet Protocol (IP) network multipathing software on Sun Cluster is called to fail over the defective interface to another adapter in the group.

Index

disks
  device groups, 24
  failure fencing, 22
  global devices, 23
  local, 23
  management, 12
  mirroring, 12
  multihost, 11, 23, 24, 35
  quorum, 20
drivers, *See* devices, IDs (DIDs)

E
environment
  hardware, 29
  software, 30

F
failfast, 22
failover
  data services, 26
  provision by Oracle Parallel Server/Real
    Application Clusters software, 27
  services, 10
  transparent, 10
failure
  detection, 13
  fencing, 22
  hardware and software, 13
fault tolerance, 9
fencing, 22
file locking, 33
file systems
  clusters, 12, 32
  mounting, 32

G
global devices
  description, 23
  disk device groups, 24
  mounting, 32
global namespace, 23

H
hardware
  cluster interconnects, 35
  cluster nodes, 17
  environment, 29
  failure, 13
  high availability, 9
  redundant array of independent disks
    (RAID), 12
  Sun StorEdge Traffic Manager, 12
high availability, 9

I
IDs, device, 23
interconnects, *See* cluster, interconnects
interfaces, 20, 36, 37
Internet Protocol (IP), 27
IP network multipathing, 11, 20, 36
IPMP
  See IP network multipathing

L
load balancing
  description, 33
  policies, 34
local devices, 24
logical hostname, failover data services, 26

M
membership, 17, 18, 31
monitoring
  disk paths, 20
  failure, 13
  network interfaces, 20
mounting, 32
multihost storage, 11
multipathing, 11, 13, 20, 36

V
VERITAS Volume Manager (VxVM), 12
volume management, 12, 35
vote count, quorum, 20