ORACLE
SOLARIS

An Oracle White Paper
July 2011

RUNNING ORACLE® REAL APPLICATION CLUSTERS ON ORACLE SOLARIS ZONE CLUSTERS

Deploying Oracle RAC on Oracle Solaris Zone Clusters

Virtualization technologies are a popular means to consolidate multiple applications onto a single system for better system utilization. Oracle® Solaris Cluster provides Oracle Solaris Zone Clusters (also called Oracle Solaris Containers Clusters), which provide virtual clusters and support the consolidation of multiple cluster applications onto a single cluster. Specifically, this paper describes how Oracle Real Application Clusters (RAC) can be deployed on a zone cluster.

This paper addresses the following topics:

- "Zone Cluster Overview" provides a general overview of zone clusters.
- "Benefits of Using Oracle Solaris Cluster" provides a summary of the benefits that using Oracle Solaris Cluster bring to Oracle RAC when deployed in zone clusters
- "Oracle RAC in Zone Clusters" describes how zone clusters work with Oracle RAC.
- "Example: Zone Clusters Hosting Oracle RAC" steps through an example configuring Oracle RAC on a zone cluster.
- "Oracle RAC Configurations" provides details on the various Oracle RAC configurations supported on zone clusters.

This paper assumes familiarity with basic Oracle Solaris Cluster and zone cluster concepts. Refer to the Oracle paper *Zone Clusters—How to Deploy Virtual Clusters and Why* for a more detailed introduction to zone clusters.

Zone Cluster Overview

Two types of clusters can be configured with the Oracle Solaris 10 operating system: *global clusters* and *zone clusters*.

- **Global cluster.** The global cluster contains all global zones in a collection of Oracle Solaris hosts. The global cluster can optionally contain non-global zones with no membership votes. Only the global zone in the global cluster has the privileges needed to perform administrative actions for a zone cluster.
- **Zone cluster.** A zone cluster is composed of one or more non-global zones, which are all of zone brand type *cluster*. Each cluster node of a zone cluster resides on a different Oracle Solaris host.

A zone cluster node requires that the global zone on that same Oracle Solaris host must be booted in cluster mode in order for the zone cluster node to be operational. All zone cluster nodes must be on Oracle Solaris hosts belonging to the same global cluster. The zone cluster nodes can be a subset of Oracle Solaris hosts for that same global cluster. While a zone cluster depends upon a global cluster, a global cluster does not depend upon any zone cluster.

Figure 1 shows a four-machine hardware configuration supported by the Oracle Solaris Cluster product. Figure 2 shows multiple clusters running on that same four-machine hardware configuration.

An important point is that a cluster node is a cluster software construct that does not necessarily have a one-to-one relationship to hardware; a single machine can host multiple cluster nodes.

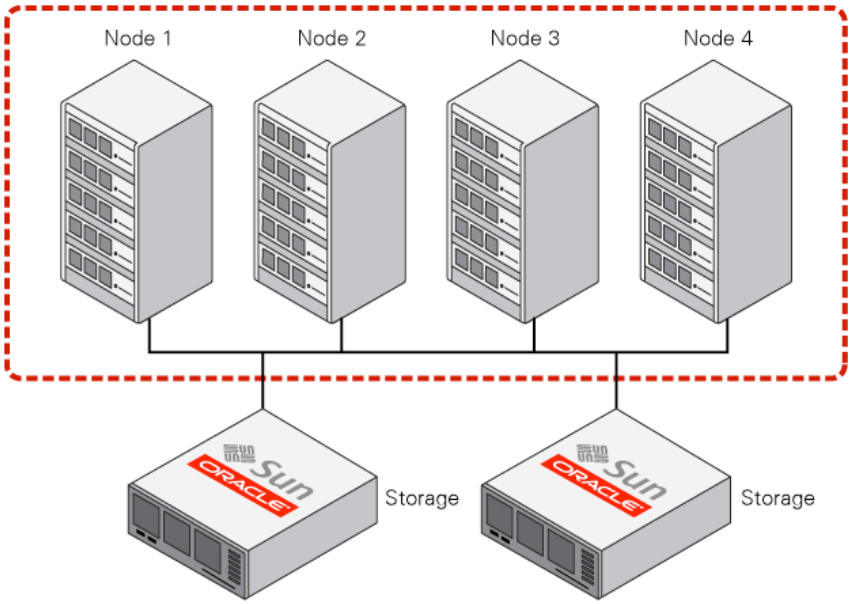

Figure 1. A four-machine hardware configuration for clustering.

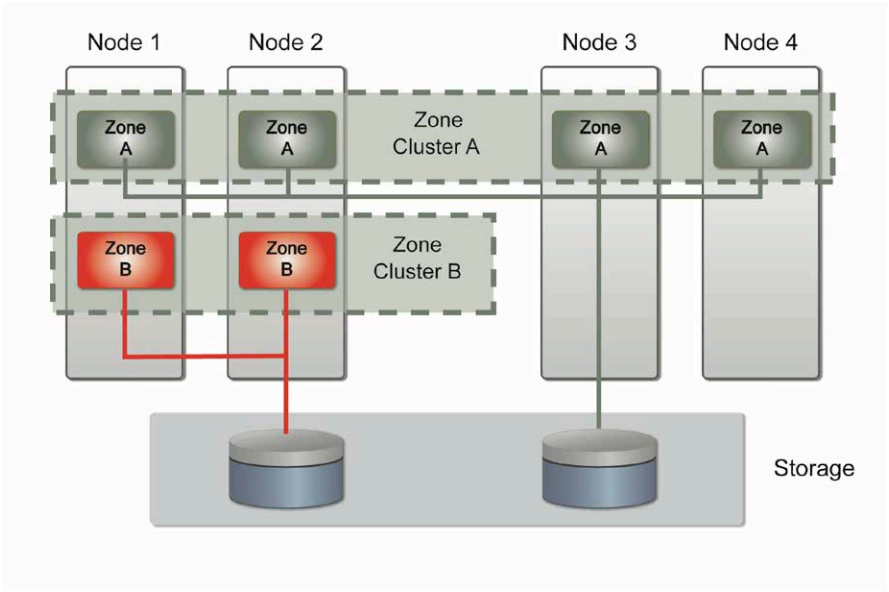

Figure 2. Multiple clusters deployed on a four-machine configuration.

The zone cluster can be viewed as a virtual cluster. The zone cluster presents the illusion to a cluster application that the zone cluster is the traditional cluster.

Zone clusters provide the following features:

- **Security isolation.** Applications and users within one zone cluster cannot see or access data or applications running in a different zone cluster.
- **Application fault isolation.** Actions taken by applications within a zone cluster, such as issuing a reboot command, do not affect applications outside of that zone cluster.
- **Resource management.** Resources, or shares of resources, can be allocated to a zone cluster. This includes resources such as CPUs, memory, devices, and file systems.
- **Dedicated cluster model.** Zone clusters provide the illusion that the zone cluster is a traditional cluster dedicated for the use of cluster applications within that zone cluster. The applications are not aware of any other clusters on that hardware. The traditional cluster can be thought of as the kind of cluster that ran on machines before virtual machines and zones existed.

One important reason for deploying a zone cluster is to consolidate multiple cluster applications onto a single physical cluster configuration. Consolidation often means that applications belonging to multiple organizations can now run on the same hardware. For example, the Oracle RAC databases for finance, sales, and engineering organizations can be placed on the same hardware using zone clusters. Zone clusters also support the consolidation of a wide range of applications. For example, the front-end, application, and database tiers of the traditional datacenter can all be consolidated onto one set of hardware using multiple zone clusters.

The database is a critical application for many organizations. In order to ensure reliability, many organizations separate production, test, and development onto separate hardware. Zone clusters enable organizations to consolidate test and development functions on one hardware configuration. The engineers can all work concurrently, and time slicing can be eliminated. The administrators can dynamically create and destroy zone clusters without impacting other zone clusters.

Zone clusters can support upgrades of the databases in an incremental fashion. For example, it is possible to run Oracle RAC 9i, RAC 10g, and RAC 11g in separate zone clusters concurrently on the same hardware. Administrators can now upgrade one database at a time to new data software releases at their own convenience.

Many software vendors charge license fees for software based on the number of CPUs. The administrator can control the number of CPUs in a zone cluster, and hence control the number of CPUs available to the software. Some vendors use this number of CPUs as the basis for the license fee. Many organizations run applications in conjunction with the database. The database software can be placed in a separate zone cluster from the other applications, and the administrator can control the number of CPUs separately for these diverse software systems. Substantial cost savings can result from this approach.

Note – Refer to the Oracle paper, "*Zone Clusters— How to Deploy Virtual Clusters and Why,*" for detailed examples of these use cases and their associated cost savings.

Benefits of Using Oracle Solaris Cluster

When the Oracle Solaris Cluster software is deployed on a collection of machines, it brings benefits to applications deployed on both the global zones of those machines and the zone clusters which are configured. These benefits are available to both the Oracle RAC databases deployed and the other software components that make up the complete application stack.

These features include:

- A robust, kernel-based membership mechanism
- A reliable storage fencing mechanism
- A pseudo STREAM network device driver (`clprivnet`) that delivers highly available, trunked connectivity through a single network address
- A consistent disk device namespace across all cluster machines
- A mature, fully-featured framework for controlling application components
- An extensive portfolio of agents to control and monitor application components
- A platform for building disaster recover solutions that encompass the entire application stack

These features are outlined in the sections below. A more detailed discussion can be found in the book *Oracle Solaris Cluster Essentials*.

Cluster Membership

Oracle Solaris Cluster membership is determined by a kernel-based membership monitor. Being kernel based avoid almost all of the problems that a similar framework can suffer when implemented as a set of user processes.

The membership monitor uses the networks that comprise the private interconnects to send heartbeat messages between the machines in the cluster. Failure of a cluster machine is detected when no heartbeat messages have been received on any of the private interconnects for a period of 10 seconds. This event triggers a cluster reconfiguration which results in the new membership information being propagated to any zone clusters configured on the remaining cluster machines. As a result, Oracle RAC deployments are quickly and accurately made aware of new membership information, regardless of whether they are deployed in the global cluster or in a zone cluster. Machines that should not participate in the new cluster are ejected from the cluster by being forcible terminated. This also terminates any zone cluster nodes present on the machine.

In addition to providing the new membership information, the membership monitor also triggers the storage fencing process, to protect against data corruption, and the quorum mechanism, which protects against cluster partitions in both space (“split-brain”) and time (“amnesia”)

Storage Fencing

In some cases, a machine may have been ejected from a cluster because it is in an errant state and has failed to communicate with its peers within the allotted time span. In such cases, it is important to ensure that it cannot continue writing to shared storage after the cluster reconfiguration process has started. The SCSI fencing mechanisms employed by Oracle Solaris Cluster ensures that any attempt to do so by the ejected node, or nodes, results in them being panicked immediately.

Oracle Solaris Cluster supports fencing of NAS units, disks, and RAID devices. This very important data integrity feature, which prevents data corruption, has been supported by Oracle Solaris Cluster for many years and is also supported by zone clusters.

`clprivnet` Pseudo STREAM Network Driver

Oracle Solaris Cluster provides improved communications between nodes through the private interconnect, using the `clprivnet` driver. There can be up to six private interconnects. Oracle Solaris Cluster stripes traffic across all available communication paths when using the `clprivnet` driver. Oracle Solaris Cluster further transparently recovers from communication path failures and guarantees delivery as long as at least one path remains operational. This driver can be used by Oracle 9i, Oracle 10g, Oracle 11gR1 and Oracle 11gR2, in each case, the clprivnet driver presents a single virtual network adapter and a single IP address to the Oracle software.

Furthermore, configuration of the necessary network addressing for this interface in the zone cluster is managed automatically when the zone cluster is created, removing the risk of misconfiguration.

Consistent Disk Device Namespace

Creating or upgrading a cluster with more than one machine type can often result in shared storage devices having different cXtYdZ device paths on each machine. Oracle Solaris Cluster automatically manages these differences by creating a consistent namespace across all machines in the cluster. These new device names can then be used either by shared QFS or Oracle's Automatic Storage Management. This feature removes the need for a system administrator to create and manage symbolic links to such devices.

Application Component Control

Oracle Solaris Cluster has an integrated Resource Group Manager that can control the start, stop and health monitoring of application components through a set of application specific agents. The Resource Group Manager is driven by events, such as nodes joining and leaving the cluster, and through administrative controls from the command line. The Resource Group Manager allows both positive and negative affinities to be defined between groups of application components, and dependencies between individual components. Both types of relationship can span zone cluster boundaries. This capability then allows application components in one tier, in one zone cluster, to be dependent on components of another tier, such as the RAC database itself, in a separate zone cluster. This is not something that can be done with the Oracle Clusterware or Grid Infrastructure framework.

Extensive Agent Portfolio

Oracle Solaris Cluster comes with an extensive portfolio of agents that can start, stop or monitor the health of an application component. Some of the agents pertinent to Oracle Clusterware and RAC are outlined below. There are also 50 or more other components that can control a broad range of Oracle application components, such as Oracle WebLogic, Oracle E-Business Suite, Siebel, Peoplesoft, SAP and Oracle Business Intelligence. All of these components are zone cluster enabled, allowing entire application stacks to be consolidated onto a single cluster of machines.

For applications without a pre-existing agent, the Generic Data Service allows start, stop and health probe scripts to be quickly integrated into a service that contains the best practices for application component restart and failover.

Building Disaster Recover Solutions

Deploying Oracle Solaris Cluster enables multiple clusters to be brought together in partnerships through the Oracle Solaris Cluster Geographic Edition software. This combination allows every tier of the application to be afforded a common disaster recovery framework that encompasses many of the popular data replication mechanisms. These mechanisms include both storage replication, host-based replication and application-based replication. Of particular relevance to Oracle RAC is the support for Oracle Data Guard by the Geographic Edition framework.

Separate instances of the Geographic Edition framework can run in any or all of the global cluster and the zone clusters configured on the physical cluster.

Oracle RAC in Zone Clusters

The zone cluster appears to be a dedicated cluster from the perspective of Oracle RAC. In other words, Oracle RAC should not see any difference between running in a zone cluster and running on a cluster without zones. In general, the Oracle RAC software that runs in the global cluster should run in the zone cluster without any changes. However, the current support matrices must be checked and the required Solaris and Oracle patchsets must be present before attempting to install the Oracle RAC software in a zone cluster.

A zone cluster can run the following Oracle releases:

- Oracle RAC 9i (9.2.0.5 and above)
- Oracle RAC 10g (10.2.0.3 and above)
- Oracle RAC 11gR1 (11.1.0.6 and above)
- Oracle RAC 11gR2 (11.2.0.2 and above)

Since zone clusters are independent environments, it is possible to run any combination of these Oracle RAC versions (9i/10g/11g) on the same Oracle Solaris Cluster, using different zone clusters for each different release.

Oracle RAC running in the global zone can support multiple databases. Similarly, Oracle RAC running in a zone cluster can also support multiple databases in that one zone cluster.

The following subsections provide more details on how zone clusters work with Oracle RAC.

Oracle Solaris Cluster/Oracle RAC Integration

Oracle Solaris Cluster does more than just provide a platform for Oracle RAC — Oracle Solaris Cluster integrates Oracle RAC with the cluster. In addition, Oracle Solaris Cluster manages not only Oracle RAC, but also manages applications working with Oracle RAC. The entire set of applications, including Oracle RAC, can be managed through Oracle Solaris Cluster. Oracle Solaris Cluster also provides information to Oracle Clusterware[1] (10gR2) or Grid Infrastructure (11gR1 and 11gR2) about important resources, such as devices and file systems. This integration enables the Oracle Solaris Cluster to more intelligently manage the entire set of applications and resources related to the database installation, for better overall availability. Throughout the remainder of this document, use of the term Clusterware should be taken to include Grid Infrastructure unless otherwise stated.

Figure 3 shows the management of resources from both the Clusterware and Oracle Solaris Cluster perspectives when the RAC database is held on a cluster file system or on raw devices. Figure 4 shows the same picture for configurations where an 11gR2 database is held on Oracle Automatic Storage Management (ASM). For 10gR2 and 11gR1 configurations, there are no Clusterware resources for ASM. Instead, Oracle Solaris Cluster provides the resources for ASM and installs proxy ASM resources under Clusterware control.

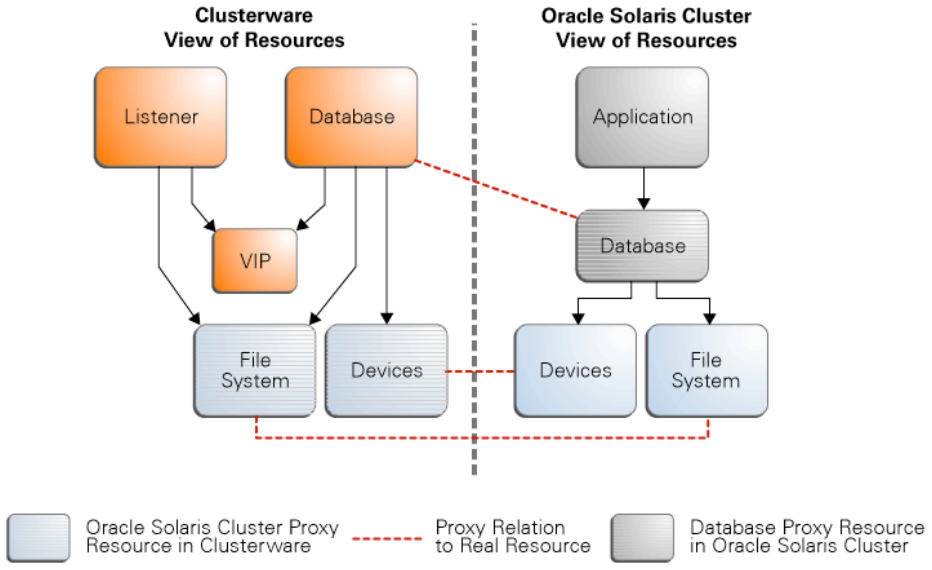

Figure 3. Oracle Solaris Cluster and Oracle Clusterware view of resources for databases held on a cluster file system or on raw devices.

[1] Oracle Clusterware was originally known as Cluster Ready Services (CRS)

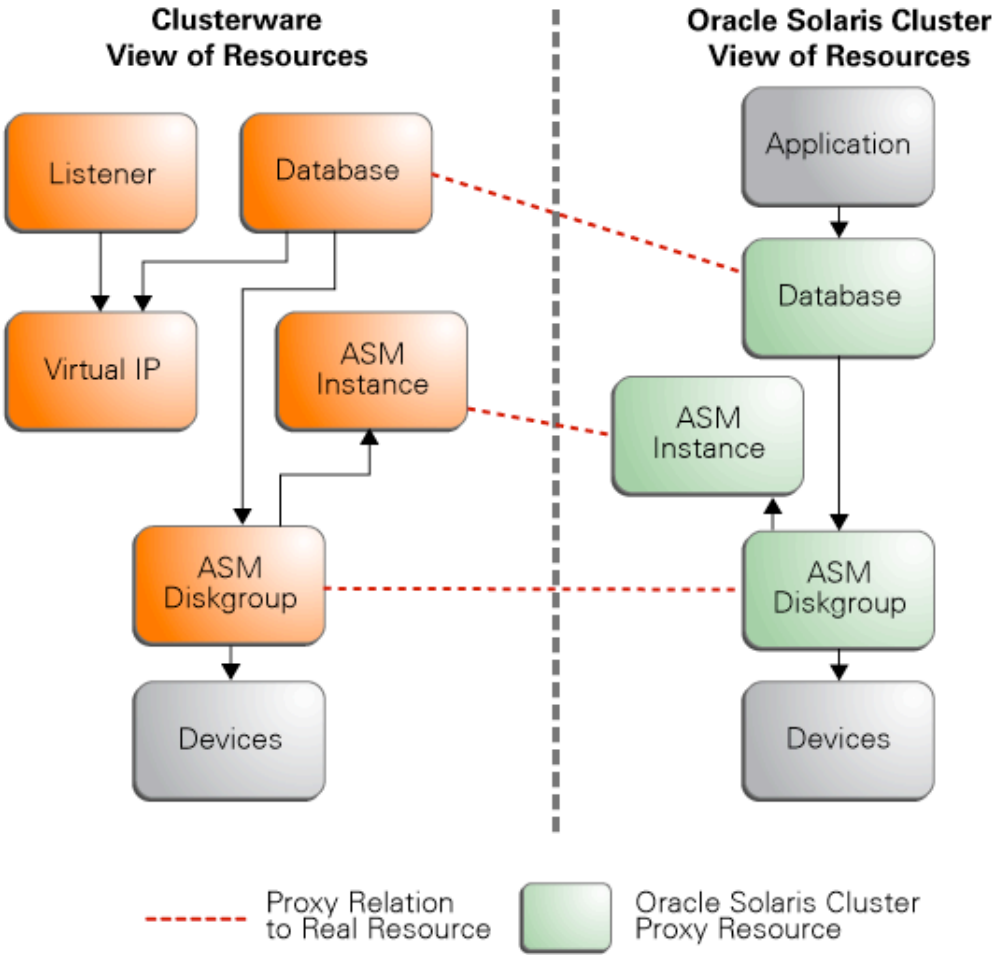

Figure 4. Oracle Solaris Cluster and Oracle Grid Infrastructure (11gR2) view of resources for databases held in ASM.

Oracle RAC Resource Isolation

Oracle RAC uses the following resources:

- A network resource (IP Address) to reach the public network
- A network resource (IP Address) to communicate with applications
- A network resource (IP Address) to communicate with other instances of the database
- Either a cluster file system or fully connected storage devices to store database information

Figure 5 shows an example of two zone clusters isolating these resources for two different Oracle RAC installations on the same physical cluster. In this example, zone cluster ZC1 supports Oracle RAC with QFS shared file systems on Oracle Solaris Volume Manager for Oracle Solaris Cluster, while zone cluster ZC2 supports Oracle RAC with Automatic Storage Management (ASM).

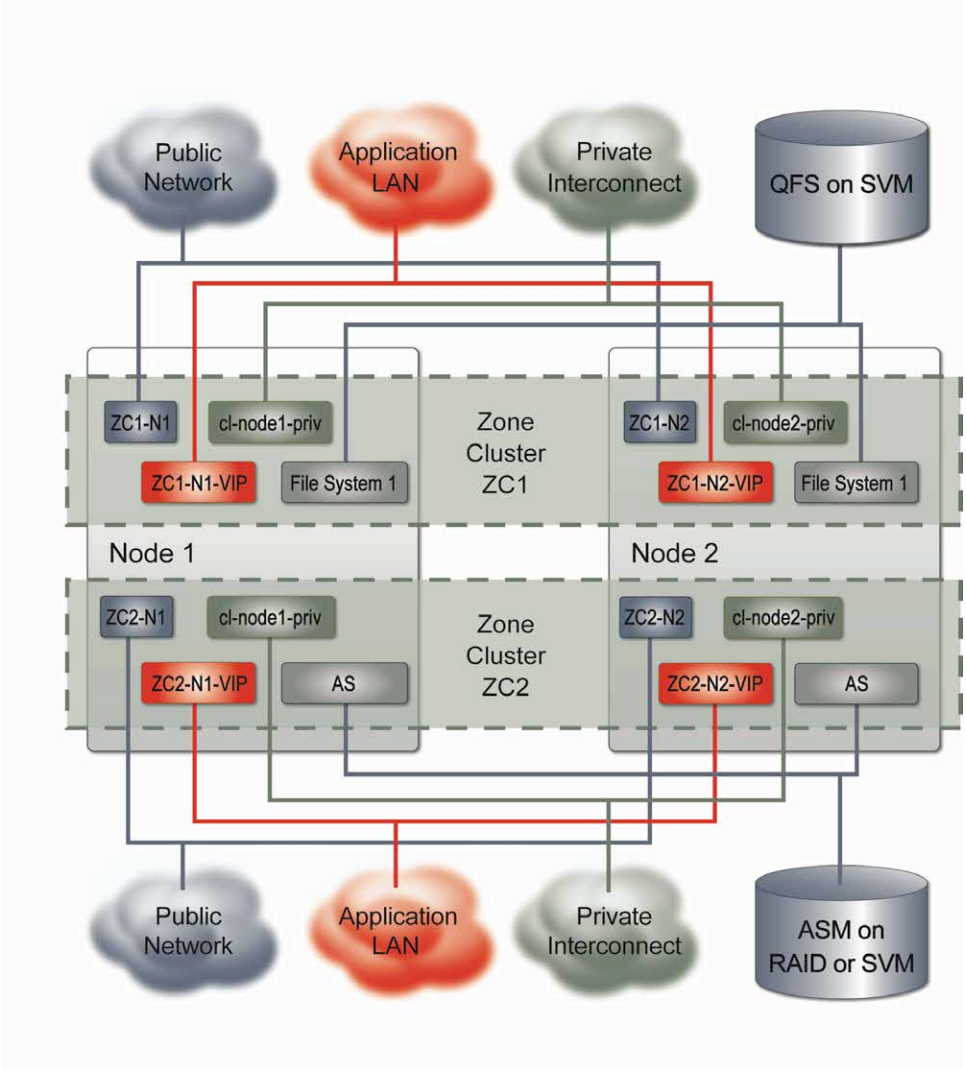

Figure 5. Oracle RAC resource isolation in zone clusters.

Common Resource Components

A number of resource components are present in all configurations supporting Oracle RAC. These common components are identified in this section to reduce duplicate descriptions. Resource components that appear only in specific configurations are described in later sections describing those configurations.

RAC Framework

The scalable RAC framework resource (`SUNW.rac_framework`) provides services to support Oracle RAC. The RAC framework resource provides platform information, such as membership, to Oracle RAC; and orchestrates Oracle RAC-related software reconfigurations. If the RAC framework resource

needs to be restarted then the global cluster or the zone cluster, depending on where the resource is configured, must be rebooted.

VUCMM Framework

The VUCMM framework resource (`SUNW.vucmm_framework`) represents the framework that enables different clustered volume managers in a Oracle Solaris Cluster configuration. This resource enables the framework status to be monitored. If the VUCMM framework resource needs to be restarted then the global cluster or the zone cluster, depending on where the resource is configured, must be rebooted.

CRS Framework

The scalable CRS framework resource (**`SUNW.crs_framework`**) monitors Oracle Clusterware and presents the state of Clusterware to the rest of the cluster. The Clusterware framework resource disables Clusterware autostart property (if it is re-enabled, it is not disable again) so that it can order Clusterware to both start and halt. However, the Clusterware resource does not re-start Clusterware after a failure because Clusterware has its own built-in features to do this. This Clusterware resource supports Oracle RAC 10g /11g. When Oracle RAC is configured in the zone cluster, the CRS framework resource resides within the zone cluster.

UNIX Distributed Lock Manager (UDLM)

The scalable UDLM resource (**`SUNW.rac_udlm`**) manages the Oracle UDLM, which provides a distributed lock manager service that supports Oracle RAC. The UDLM resource is configured in the zone cluster when supporting Oracle RAC in the zone cluster. UDLM runs only on SPARC® processor-based machines.

RAC Instance Proxy Resource

The scalable RAC instance proxy resource (**`SUNW.scalable_rac_server_proxy`**) has callbacks from Oracle Clusterware that deliver information about the state of the database. The RAC instance proxy resource receives information from Clusterware about the state of the database, which can be on either a file system or a storage device. The RAC instance proxy resource can start or stop the Oracle database via Oracle Clusterware. This resource exists in Oracle RAC 10g /11g. When Oracle RAC is configured in the zone cluster, the RAC instance proxy resource resides within the zone cluster.

RAC Server Resource

The scalable RAC server resource (**`SUNW.scalable_rac_server`**) monitors the Oracle database server. The RAC server resource can start or stop the Oracle database. This resource is only present in RAC 9i. When Oracle RAC is configured in the zone cluster, the RAC server resource resides within the zone cluster.

RAC Listener Resource

The scalable RAC listener resource (`SUNW.scalable_rac_listener`) starts and stops the Oracle listener process. This resource is only present in RAC 9i. When Oracle RAC is configured in the zone cluster, the RAC listener resource resides within the zone cluster.

ASM Instance Proxy Resource

The scalable ASM instance proxy resource (`SUNW.scalable_oracle_asm_instance_proxy`) is a proxy resource and exists only in Oracle RAC 10g and 11g configurations. The proxy resource acts as a substitute for a resource that is managed by Oracle Clusterware and enables Oracle Solaris Cluster utilities to manage Oracle RAC server instances through Oracle Clusterware. In this way, the `SUNW.scalable_asm_instance_proxy` resource type enables the clustering frameworks that are provided by Oracle Solaris Cluster and Oracle Clusterware to inter-operate. The ASM instance proxy resource can start or stop the ASM database via Oracle Clusterware and has a monitor that obtains the status information from the Clusterware resource for which the `SUNW.scalable_asm_instance_proxy` resource is acting as a proxy. When Oracle RAC is configured in the zone cluster, the ASM instance proxy resource resides within the zone cluster.

ASM Disk Group Resource

The ASM disk group resource (`SUNW.oracle_asm_diskgroup`) is used with Oracle 10g and 11g release 1 only. The resource represents a single or clustered instance Oracle ASM disk group in an Oracle Solaris Cluster configuration. Each `SUNW.scalable_asm_diskgroup` resource represents a clustered Oracle ASM disk group. Each clustered Oracle ASM disk group is uniquely identified by the value of the `asm_diskgroups` extension property on the node where the instance is running. When Oracle RAC is configured in the zone cluster, the ASM diskgroup resource resides within the zone cluster.

ASM Diskgroup Proxy Resource

The scalable ASM diskgroup proxy resource (`SUNW.scalable_asm_diskgroup_proxy`) is used with Oracle 11g release 2 only. The resource represents a single or clustered instance Oracle ASM disk group in an Oracle Solaris Cluster configuration. Each `SUNW.scalable_asm_diskgroup_proxy` resource represents a clustered Oracle ASM disk group. Each clustered Oracle ASM disk group is uniquely identified by the value of the `asm_diskgroups` extension property on the node where the instance is running. When Oracle RAC is configured in the zone cluster, the ASM diskgroup proxy resource resides within the zone cluster.

Wait for Zone Cluster Boot Resource

This scalable resource (`SUNW.wait_zc_boot`) waits for the completion of the boot of the specified zone cluster zone on each machine before the scalable resource on that same machine completes its boot method. This resource is used to ensure that certain resources do not come online until after the zone of the zone cluster boots. Specifically, this resource is used to ensure that the zone of the zone cluster boots before the system mounts the QFS shared file systems for that zone.

The following example uses the `SUNW.wait_zc_boot` resource to ensure that one QFS shared file system mounts after the zone of the zone cluster boots. Note that all of these resources and resource groups reside in the global zone.

The following commands assume that the resource groups have already been created in the global zone, and that the name of the zone cluster is `zc-rac`:

```
# clresourcetype register SUNW.wait_zc_boot
# clresource create -t SUNW.wait_zc_boot \
-g scal-wait-zc-rg \
-p ZCName=zc-rac \
wait-zc-rs
# clresource create -t SUNW.qfs \
-g mds-rg \
-p resource_dependencies=wait-zc-rs \
-p QFSFileSystem=/zone/zc-rac/root/db_qfs/Data \
mds-rs
#
```

- The first command, `clresourcetype` register, registers the `SUNW.wait_zc_boot` resource type. This step is performed only once for all zone clusters that use this resource type.
- The next command creates a `SUNW.wait_zc_boot` resource named `wait-zc-rs` in the scalable resource group `scal-wait-zc-rg`. The resource extension property `ZCName` specifies the name of the zone cluster that the `SUNW.wait_zc_boot` resource will wait upon (`zc-rac` in this example).
- The last command creates a QFS metadata server (`SUNW.qfs`) resource named `mds-rs` in the failover resource group `mds-rg`. This example assumes a zone root path of `/zones/zc-rac`, a resource dependency upon resource `wait-zc-rs`, and a QFS shared file system mounted at `/zones/zc_rac/root/db_qfs/Data`.

Sun QFS Shared File Systems

A QFS shared file system directly performs reads and writes of data from each node. The scalable mount point resource (`SUNW.ScalMountPoint`) monitors the status of the QFS shared file system on each node. This resource resides within the zone cluster.

A QFS shared file system has a single active metadata server (MDS) per file system for the entire cluster; this MDS manages the meta information of one specific QFS shared file system. Any node can host the MDS. The MDS is a failover resource that is active on only one node at a time. In case of node failure, the system automatically activates the MDS on a surviving node. The MDS resource is the only failover resource used for Oracle RAC configurations. The QFS metadata server resource (`SUNW.qfs`) manages the QFS metadata server, and this resource exists only in the global zone.

The zone model encourages placing a file system in the zone, without granting direct access to the underlying storage device. A malicious program can issue a bad `ioctl` function call that can take down the entire machine. This is not possible when accessing data storage through a file system, which

improves overall system reliability. There is no additional overhead by not placing the underlying device in the zone.

QFS shared file systems run on top of either hardware RAID or Oracle Solaris Volume Manager, which runs only in the global zone. The Oracle Solaris Volume Manager resource (`SUNW.vucmm_svm`) controls the Oracle Solaris Volume Manager. All administrative operations on an Oracle Solaris Volume Manager disk set occur in the global zone.

UDLM and Native SKGXN Support

SKGXN (system kernel generic interface node) is an Oracle RAC interface for providing platform information to Oracle RAC. Any particular zone cluster on a SPARC processor-based machine can use UDLM or native SKGXN support. Thus, a single physical cluster can support both options. However, each zone cluster can use only one of either UDLM or native SKGXN support.

Configuration Wizard for Oracle RAC

Oracle Solaris Cluster Manager includes a wizard for configuring Oracle RAC in a zone cluster. This data service configuration wizard supports the different versions of Oracle RAC and the different storage configurations described in this document. Oracle RAC configurations involve numerous resources, resource groups, resource dependencies, and resource group affinities. The RAC data service configuration wizard knows how to properly configure all of these items and helps reduce the work load of the administrator. The Oracle RAC data service configuration wizard automatically performs discovery when possible. In addition, the Oracle RAC data service configuration wizard automatically ensures that mandatory properties are established for each configuration. This configuration wizard can save the administrator much time and effort, and help eliminate configuration errors, when setting up Oracle RAC configurations.

There are two forms of the wizard that configures Oracle RAC. The Oracle Solaris Cluster Manager provides a browser-based user interface means of configuring Oracle RAC. The `clsetup` command provides a text-based user interface means of configuring Oracle RAC. Both forms of the wizard accomplish the same thing.

Figure 6 shows a screen snapshot of one of the steps of configuring Oracle RAC in a zone cluster using the brower-based wizard. In this example, the administrator selects file system mount points for Oracle RAC.

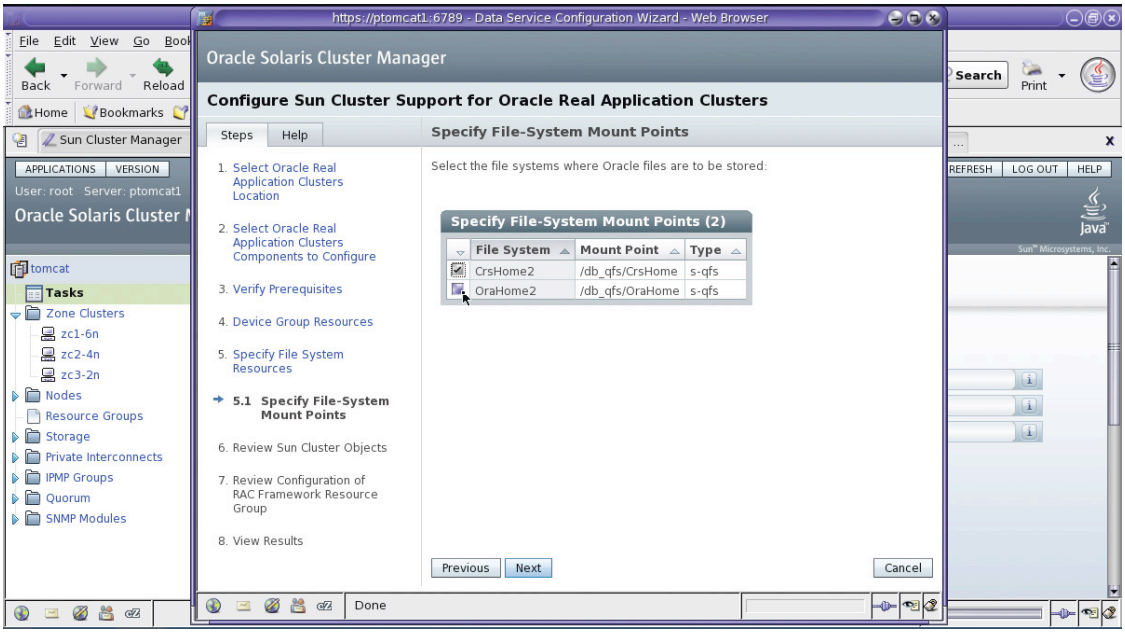

Figure 6. Configuration wizard for Oracle RAC.

Example: Zone Clusters Hosting Oracle RAC

This section provides an example of a single cluster hosting multiple Oracle RAC configurations in zone clusters. First, this section describes the cluster configuration, and then provides a step-by-step explanation of how to set up one of the zone clusters. Familiarity with Oracle Solaris, Oracle Solaris Cluster, Oracle Solaris Volume Manager, Sun QFS shared file system, and Oracle RAC administrative actions is assumed. The following discussion concentrates on those actions that are new for zone clusters.

Example Hardware Configuration

The cluster hardware consists of six Sun Fire T2000 servers, with each server having 1 GHz 6-core/24-thread UltraSPARC®T1 processors and 8 GB physical memory. Each server is configured with `e1000g1` and `e1000g5` NICs acting as private network interconnect interfaces with Jumbo-Frame enabled. In addition, each server utilizes `e1000g0` and `e1000g2` NICs as public network interfaces, and these same two NICs are configured in a single IP multipathing (IPMP) group. Two Sun StorageTek 3510 storage arrays with a total of 12 LUNs are used for disk storage.

In addition to the cluster hardware, a single Sun Fire T1000 server runs Swingbench Order Entry workloads on the databases.

This configuration is only an example and newer servers can be used for this type of deployment.

Individual Zone Cluster Configurations

This example includes three zone clusters running concurrently on the cluster hardware configuration, as shown in Figure 7:

- zc1-6n — a 6-node zone cluster hosting a 10.2.0.3 RAC database
- zc2-4n — a 4-node zone cluster hosting a 10.2.0.4 RAC database
- zc3-2n — a 2-node zone cluster hosting an 11gR1 RAC database

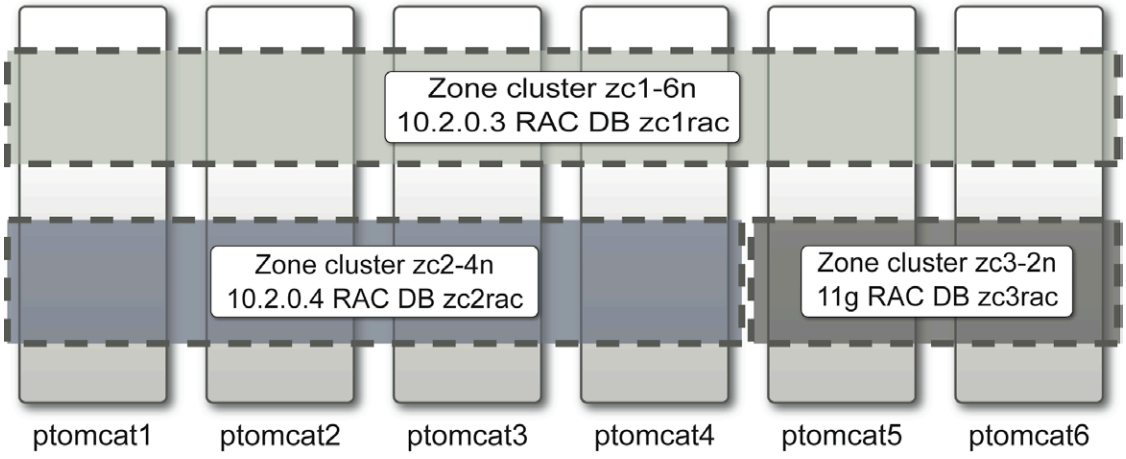

Figure 7. Overview of demo configuration.

This example configures each zone cluster node with eight dedicated CPUs and 4 GB of main memory per machine. At this time all zone clusters use the network configuration option `ip-type=shared`, which means that the zone clusters share the NICs with the global zone and each zone cluster has the use of specific IP addresses. The zone clusters share the same IPMP group for public network communications, but have different logical interfaces plumbed on the NIC.

The zone clusters similarly share the same cluster private interconnects, but have different logical interfaces plumbed on the `clprivnet` NIC.

This example grants four LUNs to each zone cluster. Each of the zone clusters uses two separate LUNs for two Sun QFS shared file systems to contain executables from Oracle. One LUN contains the Clusterware executables, and the other LUN contains the Oracle RAC database executables. Zone cluster zc1-6n uses two additional LUNs for QFS shared file systems on Oracle Solaris Volume Manager with cluster feature volumes to contain Clusterware data and the database data respectively. Zone clusters zc2-4n and zc3-2n each use additional LUNs for Oracle Solaris Volume Manager for Oracle Solaris Cluster volumes for Clusterware data and database data respectively.

Figure 8 shows the configuration for zone cluster zc1-6n. Only two of the six nodes are shown in this figure; the remaining four nodes have a similar configuration.

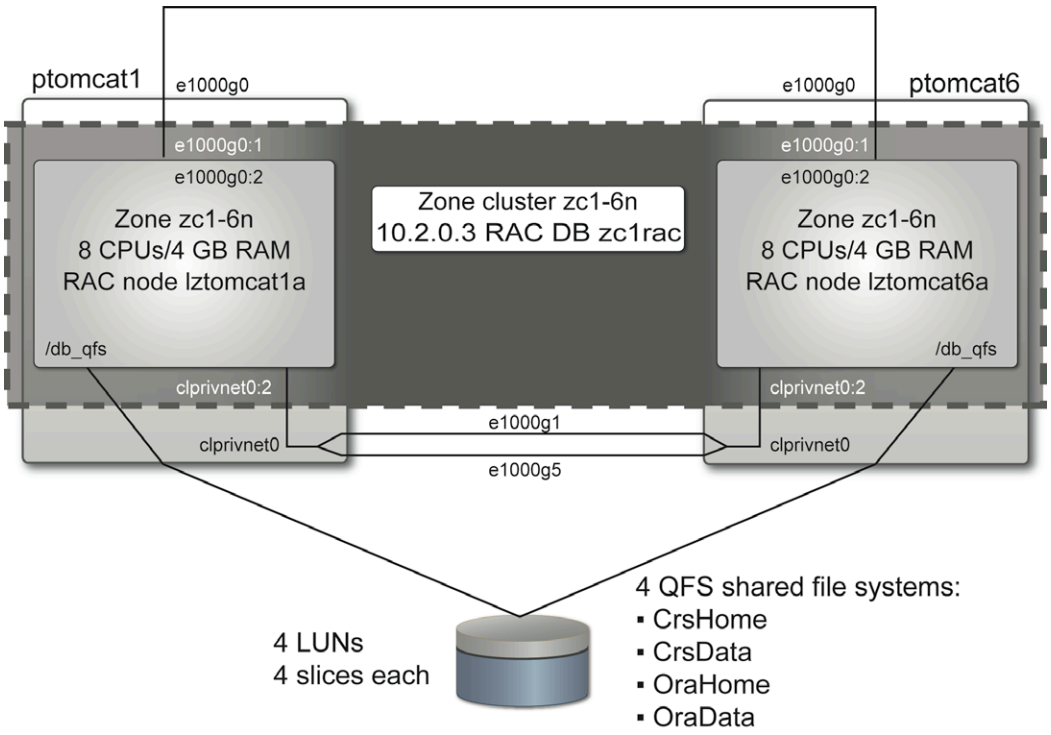

Figure 8. Zone cluster zc1-6n configuration.

Figure 9 shows the configuration for zone cluster zc2-4n. Only two of the four nodes are shown in this figure; the remaining two nodes have a similar configuration.

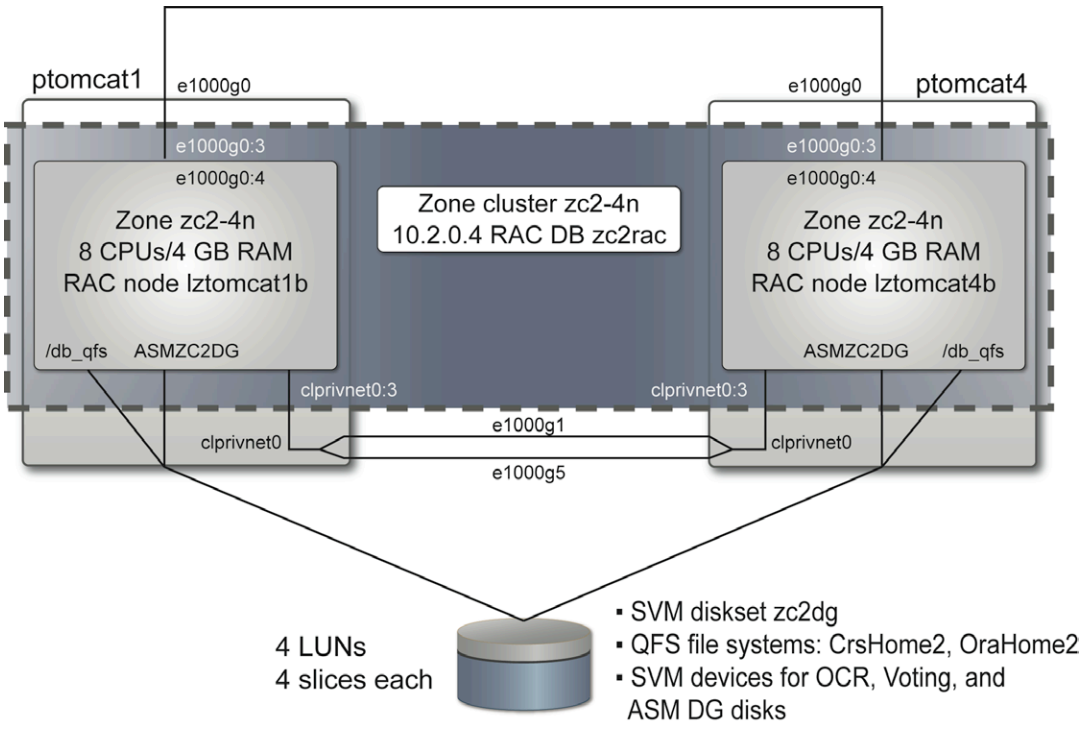

Figure 9. Zone cluster zc2-4n configuration.

Figure 10 shows the configuration for zone cluster zc3-2n.

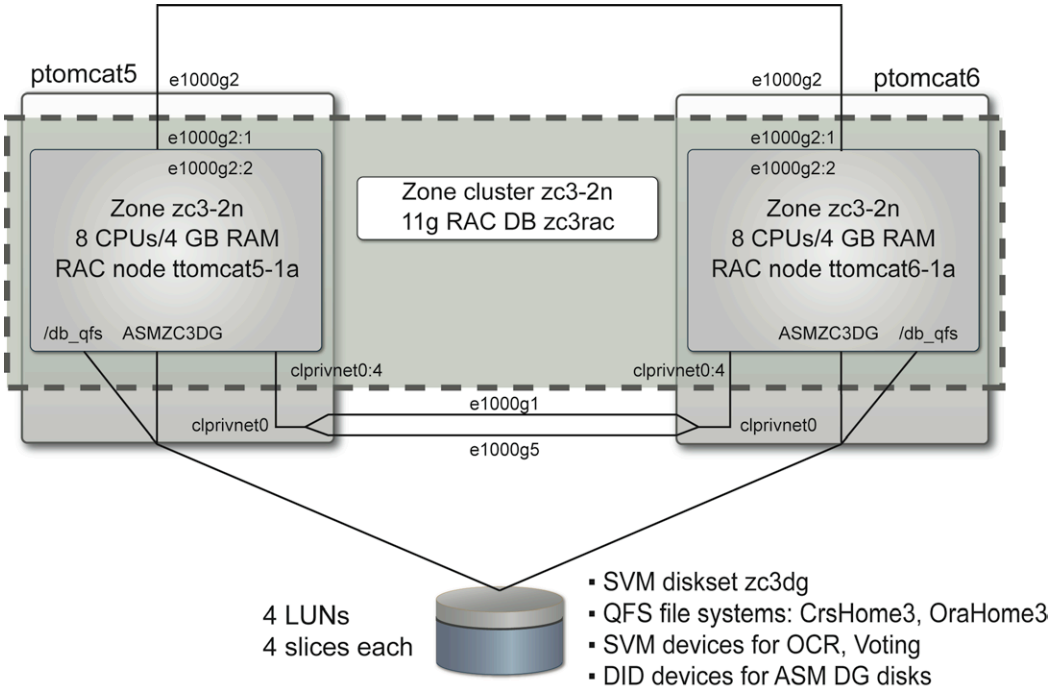

Figure 10. Zone cluster zc3-2n configuration.

This example configuration shows three different versions of Oracle RAC software running independently in different zone clusters. This example further shows zone clusters each having different numbers of nodes. In addition, two different storage topologies are used to support the Oracle RAC databases.

Installing Oracle Solaris and Oracle Solaris Cluster Software

Each machine has exactly one installation of Oracle Solaris and exactly one installation of the Oracle Solaris Cluster software. This software will support the global cluster and all zone clusters on these machines. Later when applying patches or updates, the administrator applies only one patch or update per machine for Oracle Solaris and separately for Oracle Solaris Cluster. There is no additional patch or update work required for zone clusters.

Follow the steps below to install Oracle Solaris and Oracle Solaris Cluster software.

1. First, install the Oracle Solaris 10 10/09 or later release on all machines. It is recommended that the latest patches be installed.
2. Next, install the Oracle Solaris Cluster 3.3 or later release on the machines. Again, it is recommended that the latest patches be installed.
3. As part of installing the Oracle Solaris Cluster software, the administrator configures the global cluster. The process creates the IPMP group for the public network and the `clprivnet` interface for the cluster private interconnect as part of this work. These steps are performed only one time, regardless of the number of zone clusters.

At this point the administrator is ready to prepare individual zone clusters.

Configuring the Zone Cluster

This section covers the details of how to configure the four-node zone cluster, called zc2-4n.

1. First, determine the encrypted value for the root password that will be used in this zone cluster. (Refer to the `sysidcfg` man page for more information.)
2. A storage volume or file system must exist before configuring the resources into a zone cluster. Create the Oracle Solaris Volume Manager volumes. These volumes are created in the global zone in exactly the same way as if these volumes were to be used to support the global cluster. In this example, the disk set created is named `zc2dg` and the following metadevices are created:

```
# metaset -s zc2dg -M -a -h ptomcat1 ptomcat2 ptomcat3 ptomcat4
# metaset -s zc2dg -a /dev/did/dsk/d2[0369]
# metainit -s zc2dg d11 1 1 /dev/did/dsk/d20s0
# metainit -s zc2dg d12 1 1 /dev/did/dsk/d29s0
# metainit -s zc2dg d21 1 1 /dev/did/dsk/d26s0
# metainit -s zc2dg d22 1 1 /dev/did/dsk/d23s0
# metainit -s zc2dg d31 1 1 /dev/did/dsk/d20s1
# metainit -s zc2dg d32 1 1 /dev/did/dsk/d29s1
# metainit -s zc2dg d41 1 1 /dev/did/dsk/d26s1
```

```
# metainit -s zc2dg d42 1 1 /dev/did/dsk/d23s1
# metainit -s zc2dg d51 1 1 /dev/did/dsk/d20s3
# metainit -s zc2dg d52 1 1 /dev/did/dsk/d29s3
# metainit -s zc2dg d61 1 1 /dev/did/dsk/d26s3
# metainit -s zc2dg d62 1 1 /dev/did/dsk/d23s3
# metainit -s zc2dg d71 1 1 /dev/did/dsk/d20s4
# metainit -s zc2dg d72 1 1 /dev/did/dsk/d29s4
# metainit -s zc2dg d73 1 1 /dev/did/dsk/d26s4
# metainit -s zc2dg d74 1 1 /dev/did/dsk/d23s4
# metainit -s zc2dg d1 -m d11 d12
# metainit -s zc2dg d2 -m d21 d22
# metainit -s zc2dg d3 -m d31 d32
# metainit -s zc2dg d4 -m d41 d42
# metainit -s zc2dg d50 -m d51 d52
# metainit -s zc2dg d5 -p d50 120m
# metainit -s zc2dg d60 -m d61 d62
# metainit -s zc2dg d6 -p d60 120m
# metainit -s zc2dg d7 -p d50 1m
```

3. Next, create the QFS shared file systems `CrsHome2` for Clusterware and `OraHome2` for Oracle Home on top of Oracle Solaris Volume Manager volumes. These file systems are created in exactly the same way as if these file systems were to be used to support the global cluster. These file systems have mount points that will be under the root of the zone.

   a) In the global zone of each machine ptomcat1 to ptomcat4, create or add the following content to corresponding files:

```
# more /etc/vfstab
#device    device    mount    FS     fsck    mount     mount
#to mount  to fsck   point    type   pass    at boot   options
...
CrsHome2 - /zones/zc2-4n/root/db_qfs/CrsHome  samfs – no  shared
OraHome2 - /zones/zc2-4n/root/db_qfs/OraHome  samfs – no  shared
```

```
# more /etc/opt/SUNWsamfs/mcf
...
#
# File system CrsHome2
#
CrsHome2            5      ms      CrsHome2 on     shared
/dev/md/zc2dg/dsk/d1 50     md      CrsHome2 on
/dev/md/zc2dg/dsk/d2 51     md      CrsHome2 on
#
# File system OraHome2
#
OraHome2            6      ms      OraHome2 on     shared
/dev/md/zc2dg/dsk/d3  60    md      OraHome2 on
/dev/md/zc2dg/dsk/d4  61    md      OraHome2 on
```

```
# more /etc/opt/SUNWsamfs/samfs.cmd
...
fs=CrsHome2
sync_meta=1
fs=OraHome2
sync_meta=1
```

```
# more /etc/opt/SUNWsamfs/hosts.OraHome2
ptomcat1 clusternode1-priv 1 0 server
ptomcat2 clusternode2-priv 1 0
ptomcat3 clusternode3-priv 1 0
ptomcat4 clusternode4-priv 1 0
```

```
# more /etc/opt/SUNWsamfs/hosts.CrsHome2
ptomcat1 clusternode1-priv 1 0 server
ptomcat2 clusternode2-priv 1 0
ptomcat3 clusternode3-priv 1 0
ptomcat4 clusternode4-priv 1 0
```

b) On each machine, create the mount points for the two QFS shared file systems.

```
# mkdir -p /zones/zc2-4n/root/db_qfs/CrsHome
# mkdir -p /zones/zc2-4n/root/db_qfs/OraHome
```

c) The two QFS shared file systems span the cluster node. Therefore, execute the following commands from just one machine (ptomcat1 in this example).

```
# /opt/SUNWsamfs/sbin/sammkfs -S CrsHome2
# /opt/SUNWsamfs/sbin/sammkfs -S OraHome2
```

Note – Oracle Solaris does not allow a file system to be already mounted under the root of a file system at the time the zone boots. After finishing work with these file systems, ensure that all are unmounted before continuing.

1. Create zone cluster zc2-4n using the following `clzonecluster` command.

```
# clzonecluster configure zc2-4n
clzc:zc2-4n> create
clzc:zc2-4n> set zonepath=/zones/zc2-4n
clzc:zc2-4n> set autoboot=true
```

The first line (above) assigns the name zc2-4n to the newly created zone cluster. Remaining commands specify the location of the root of the zone, and tell the software to automatically boot the zone cluster zone whenever the machines boots. This configuration information can also be placed in a command file and passed to the `clzonecluster` command.

2. Next, specify the `sysidcfg` information using the add `sysid` subcommand.

```
clzc:zc2-4n> add sysid
clzc:zc2-4n:sysid> set root_password=ZiitH.NOLOrRg
clzc:zc2-4n:sysid> set name_service="NIS{domain_name=example.com name_server=ns8-137(123.4.5.6)}"
clzc:zc2-4n:sysid> set nfs4_domain=dynamic
clzc:zc2-4n:sysid> set security_policy=NONE
clzc:zc2-4n:sysid> set system_locale=C
clzc:zc2-4n:sysid> set terminal=xterm
clzc:zc2-4n:sysid> set timezone=US/Pacific
clzc:zc2-4n:sysid> end
```

The system does not encrypt the root password — the root password must be entered in an encrypted format. The root password is the only field that must be specified. If a value for any other field is not specified, the `clzonecluster` command will provide a value based upon the corresponding value for the global zone. Although this example sets multiple values, it is anticipated that default values will be used in most cases. The `clzonecluster` command sets the `sysidcfg` information for each zone of the zone cluster, so that the administrator does not have to repeat that work on each machine.

3. Next, identify the nodes of the zone cluster using the `add node` subcommand.

```
clzc:zc2-4n> add node
clzc:zc2-4n:node> set physical-host=ptomcat1
clzc:zc2-4n:node> set hostname=lztomcat1b
clzc:zc2-4n:node> add net
clzc:zc2-4n:node:net> set address=lztomcat1b
clzc:zc2-4n:node:net> set physical=e1000g0
clzc:zc2-4n:node:net> end
clzc:zc2-4n:node> end
clzc:zc2-4n> add node
clzc:zc2-4n:node> set physical-host=ptomcat2
clzc:zc2-4n:node> set hostname=lztomcat2b
clzc:zc2-4n:node> add net
clzc:zc2-4n:node:net> set address=lztomcat2b
clzc:zc2-4n:node:net> set physical=e1000g0
clzc:zc2-4n:node:net> end
clzc:zc2-4n:node> end
clzc:zc2-4n> add node
clzc:zc2-4n:node> set physical-host=ptomcat3
clzc:zc2-4n:node> set hostname=lztomcat3b
clzc:zc2-4n:node> add net
clzc:zc2-4n:node:net> set address=lztomcat3b
clzc:zc2-4n:node:net> set physical=e1000g0
clzc:zc2-4n:node:net> end
clzc:zc2-4n:node> end
clzc:zc2-4n> add node
clzc:zc2-4n:node> set physical-host=ptomcat4
```

```
clzc:zc2-4n:node> set hostname=lztomcat4b
clzc:zc2-4n:node> add net
clzc:zc2-4n:node:net> set address=lztomcat4b
clzc:zc2-4n:node:net> set physical=e1000g0
clzc:zc2-4n:node:net> end
clzc:zc2-4n:node> end
```

The `physical-host` is the name of the corresponding global cluster node. The hostname is the name of the zone cluster node that is used when specifying where to run a cluster application in a Resource Group Manager (RGM) node list. It is recommended to also enter the hostname in the `/etc/hosts` file along with the IP address.

The `net` resource specifies the IP address that will be plumbed on the specified NIC when the zone boots. The administrator can log in to the zone cluster node using the specified IP address or hostname.

The zone cluster zone name is always the same as the name of the zone cluster, and the zone name is not the same as the hostname.

4. Oracle RAC 10g and later releases require an IP address for each node. This IP address is called a VIP address. These IP addresses are specified using the `add net` subcommand in the cluster-wide context.

```
clzc:zc2-4n> add net
clzc:zc2-4n:net> set address=lztomcat1b-vip
clzc:zc2-4n:net> end
clzc:zc2-4n> add net
clzc:zc2-4n:net> set address=lztomcat2b-vip
clzc:zc2-4n:net> end
clzc:zc2-4n> add net
clzc:zc2-4n:net> set address=lztomcat3b-vip
clzc:zc2-4n:net> end
clzc:zc2-4n> add net
clzc:zc2-4n:net> set address=lztomcat4b-vip
clzc:zc2-4n:net> end
```

Note – For Oracle RAC 9i deployments, the administrator would instead specify IP addresses that would be used for Logical Hosts.

5. Next, information about the two QFS shared file systems that will contain CrsHome2 and OraHome2, `/db_qfs/CrsHome` and `/db_qfs/OraHome` respectively, is added to the zone cluster configuration. Note that in releases prior to 11gR2, shared QFS can be used for Oracle Clusterware installation, while with 11gR2, shared QFS cannot be used for Oracle Grid Infrastructure installation. But in all releases, shared QFS can be used for for Oracle RAC RDBMS installation.

```
clzc:zc2-4n> add fs
clzc:zc2-4n:fs> set dir=/db_qfs/CrsHome
clzc:zc2-4n:fs> set special=CrsHome2
clzc:zc2-4n:fs> set type=samfs
clzc:zc2-4n:fs> end
clzc:zc2-4n> add fs
clzc:zc2-4n:fs> set dir=/db_qfs/OraHome
clzc:zc2-4n:fs> set special=OraHome2
clzc:zc2-4n:fs> set type=samfs
clzc:zc2-4n:fs> end
```

When using a file system, the best practice is to configure just the file system into the zone cluster. If the storage device is not explicitly configured in the zone cluster, then no one can execute potentially dangerous `ioctl` operations on the storage device. The file system operates correctly with no performance penalty when the storage device is not configured into the zone cluster.

Note that the `dir` property specifies the mount point relative to the zone cluster root path. This will be how the files systems are accessed from inside the zone cluster. The zone cluster zc2-4n uses Oracle ASM on top of operating system volumes, which in this case are volumes provided by Oracle Solaris Volume Manager with cluster feature. ASM can also be configured with DID devices.

Note – When Oracle RAC is run in a zone cluster, the `inherit-pkg-dir` directive for the `/opt` file system cannot be used. Oracle RAC requires the ability to both read and write the `/opt` file system in the zone cluster. Each zone cluster must have its own `/opt` file system in order to prevent conflicts.

6. The zone cluster zc2-4n uses Oracle Solaris Volume Manager mirrored `d5`, `d6`, and `d7` volumes as the Oracle Cluster Registry (OCR) and Voting disks. ASM uses non-mirrored Oracle Solaris Volume Manager devices `d71-74`. (If mirrored devices are desired, the administrator would configure ASM to do mirroring between those devices.)

```
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/zc2dg/rdsk/d5"
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/zc2dg/rdsk/d6"
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/zc2dg/rdsk/d7"
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/zc2dg/rdsk/d71"
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/zc2dg/rdsk/d72"
```

```
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/zc2dg/rdsk/d73"
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/zc2dg/rdsk/d74"
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/shared/1/rdsk/d5"
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/shared/1/rdsk/d6"
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/shared/1/rdsk/d7"
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/shared/1/rdsk/d71"
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/shared/1/rdsk/d72"
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/shared/1/rdsk/d73"
clzc:zc2-4n:device> end
clzc:zc2-4n> add device
clzc:zc2-4n:device> set match="/dev/md/shared/1/rdsk/d74"
clzc:zc2-4n:device> end
```

7. Oracle Solaris provides a set of privileges to all non-global zones. The `set limitpriv` command can be used to specify additional privileges. Oracle RAC requires privileges that are not provided by default, and these privileges are set with the following command.

```
For 9i, 10gR2 and 11gR1 use:
clzc:zc2-4n> set limitpriv="default,proc_priocntl,proc_clock_highres"
For 11gR2 use:
clzc:zc2-4n> set limitpriv="default,proc_priocntl,proc_clock_highres,sys_time"
```

   The `proc_lock_memory` privilege grants the ability to lock down memory. The database instance locks the configured SGA memory. This privilege is already present in the zone cluster by default.

8. Allocate CPU resources. Resource control is an important part of cluster application consolidation. By specifying the number of CPUs, the administrator can control the software license fees for the application in the zone cluster. The Oracle Solaris OS supports multiple approaches for managing the CPUs in a zone cluster. The fair share scheduler can be used to assign relative shares of CPUs when license fees are not a consideration. The `importance` parameter establishes a weight for the use of the Oracle Solaris OS in deciding which zone has more importance when assigning CPUs.

```
clzc:zc2-4n> add dedicated-cpu
clzc:zc2-4n:dedicated-cpu> set ncpus=8
clzc:zc2-4n:dedicated-cpu> set importance=20
clzc:zc2-4n:dedicated-cpu> end
```

9. Allocate memory resources. It is recommend that the amount of memory be explicitly controlled for zone clusters supporting databases. When setting memory limits, remember to allow space for both the executables and the data. The following commands explicitly specify the amount of memory to be used by this zone cluster.

```
clzc:zc2-4n> add capped-memory
clzc:zc2-4n:capped-memory> set physical=4G
clzc:zc2-4n:capped-memory> set swap=5G
clzc:zc2-4n:capped-memory> set locked=4G
clzc:zc2-4n:capped-memory> end
```

10. The following commands are used to indicate that the configuration is complete.

```
clzc:zc2-4n> commit
clzc:zc2-4n> exit
```

11. At this point, zone cluster zc2-4n has been configured. The following command now installs the system software into the zone cluster.

```
# clzonecluster install zc2-4n
```

    The system installs all zones of the zone cluster in parallel, but the installation can take substantial time.

12. Zone cluster configuration and system software installation are now complete. The following command boots all zone cluster nodes.

```
# clzonecluster boot zc2-4n
```

    The first boot of the zone cluster performs initialization work, which includes setting the `sysidcfg` information. After initialization, the zone cluster automatically reboots.

13. The zone cluster zc2-4n is now up. The status of the zone cluster can be viewed using the Oracle Solaris Cluster Manager Graphical User Interface (GUI), also called a browser-based user interface.

Figure 11 shows the Oracle Solaris Cluster Manager output when examining the status of zone cluster zc2-4n.

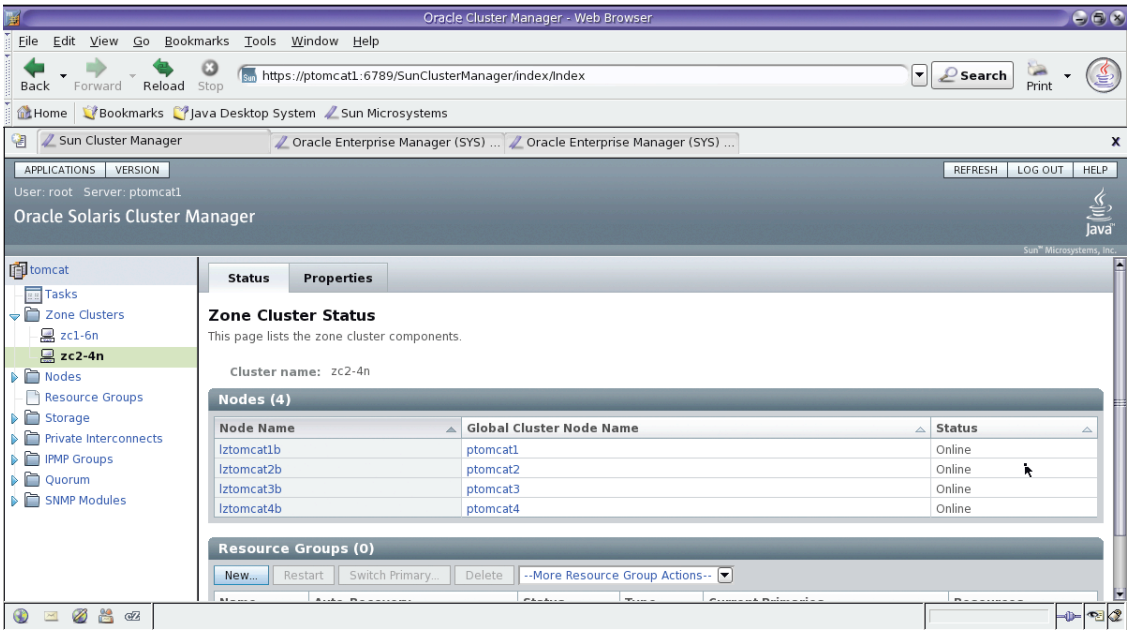

Figure 11. Screen shot — zone cluster status.

Configuring the Oracle Databases

Zone cluster zc2-4n is now ready to support applications, but no applications have been configured yet. Oracle RAC databases can be manually installed and configured in a zone cluster. But, in order to make this process easier, Oracle Solaris Cluster provides a data service configuration wizard for Oracle RAC in a zone cluster. The configuration wizard understands the requirements of running Oracle RAC in a zone cluster and can reduce the number of steps that must be performed by the administrator. The configuration wizard automatically discovers resources available in the zone cluster and presents them at the appropriate step in the process. Use of the configuration wizard is strongly recommended.

1. If UDLM is being used, the administrator first creates an Oracle user account and then installs the UDLM software.

Note – When SKGXN is used, no action is required at this point. The SKGXN software comes with the rest of the Oracle RAC software.

2. Start the Oracle Solaris Cluster Manager GUI.

   The Oracle Solaris Cluster Configuration Wizard for Oracle RAC supports all versions of Oracle RAC that Oracle Solaris Cluster supports, and can configure all supported storage topologies. The wizard is included as part of the Oracle Solaris Cluster Manager GUI. Figure 12 shows the initial screen snapshot of the wizard which selects zone cluster zc2-4n.

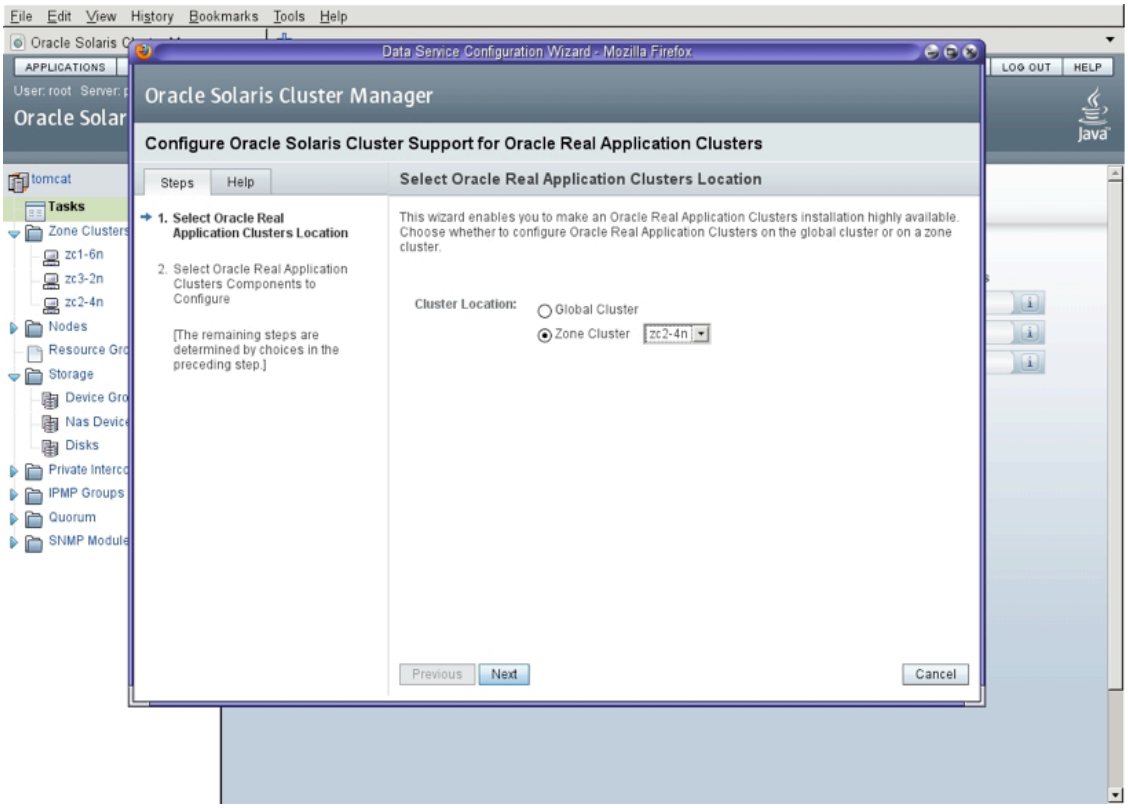

Figure 12. Screen shot — select Oracle RAC location.

The Oracle Solaris Cluster Manager GUI, along with all of the wizards that run under the Oracle Solaris Cluster Manager, runs in the global cluster in a global zone.

3. The Oracle Solaris Cluster Configuration Wizard for RAC then proceeds to the next step, which identifies six configuration areas:

   - RAC framework resource group
   - Multi-owner Volume-Manager Framework Resource Group
   - Storage resources for Oracle files
   - Oracle Clusterware Framework Resource
   - Automatic Storage Management
   - Resources for Oracle RAC database instances

   Each of these areas constitute a kind of wizard for a portion of the needed work. Each of these sub-wizards walks the administrator through a series of screens that explain what is needed at that point, display possible options, and ask the administrator to make decisions.

At nearly the end of this process, each sub-wizard displays the selected choices. The administrator can modify some choices at this point. In some cases, it is necessary to backtrack to modify other choices. When satisfied, the administrator approves the selections. The wizard then executes the appropriate commands and displays both the commands and the final result on the next screen.

In subsequent steps, this example walks through some of the different sub-wizards.

4. The first sub-wizard is the RAC Framework Resource Group sub-wizard. This sub-wizard must be executed first, and is executed once per zone cluster. Use the RAC Framework Resource Group sub-wizard to select the zone cluster nodes that will host the Oracle RAC database and choose between supporting Oracle RAC with SKGXN or UDLM. This sub-wizard configures the RAC framework resource group and the resources contained in this resource group.

   Figure 13 shows a screen snapshot from the RAC Framework Resource Group sub-wizard that lists the steps for this sub-wizard.

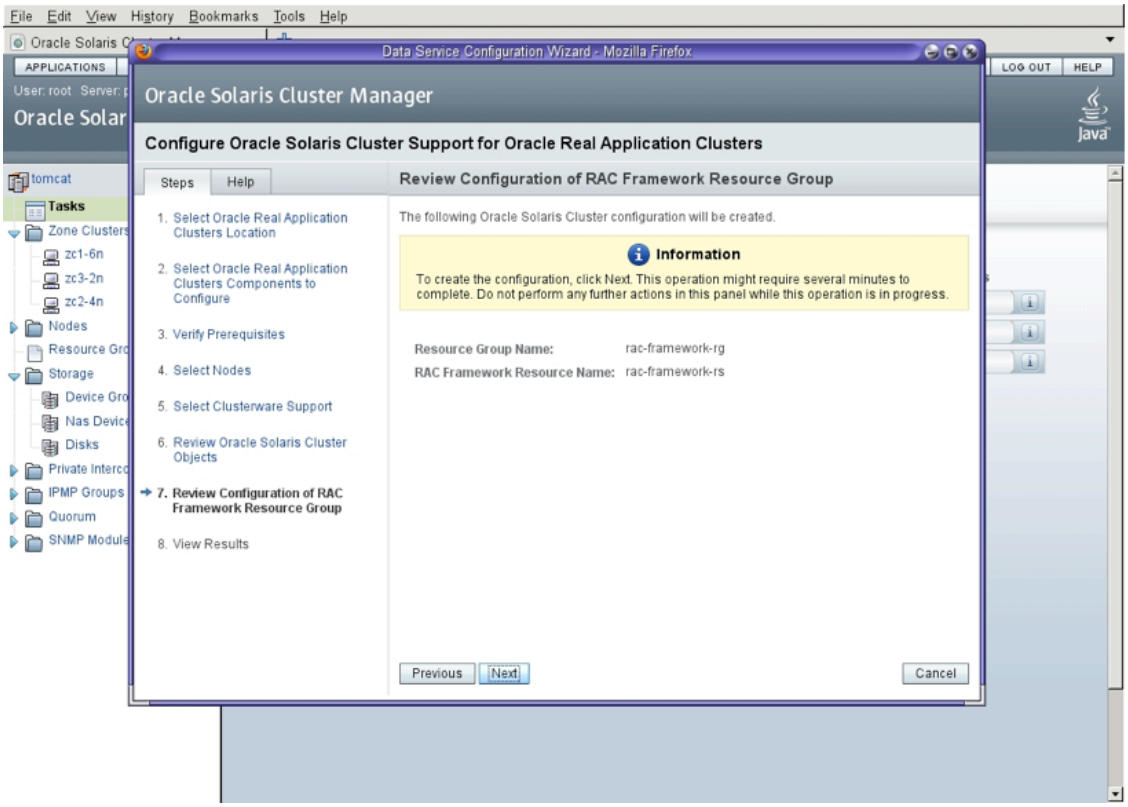

Figure 13. Screen shot — RAC Framework Resource Group sub-wizard.

5. Next, the Multi-owner Volume-Manager Framework Resource Group sub-wizard is executed. This sub-wizard can be executed before, or after, the RAC Framework Resource Group sub-wizard, but must be executed before the Storage Resources for Oracle Files sub-wizard. This sub-wizard is responsible for creating the resources needed to support the selected volume manager. Figure 14 shows a screenshot for the sub-wizard.

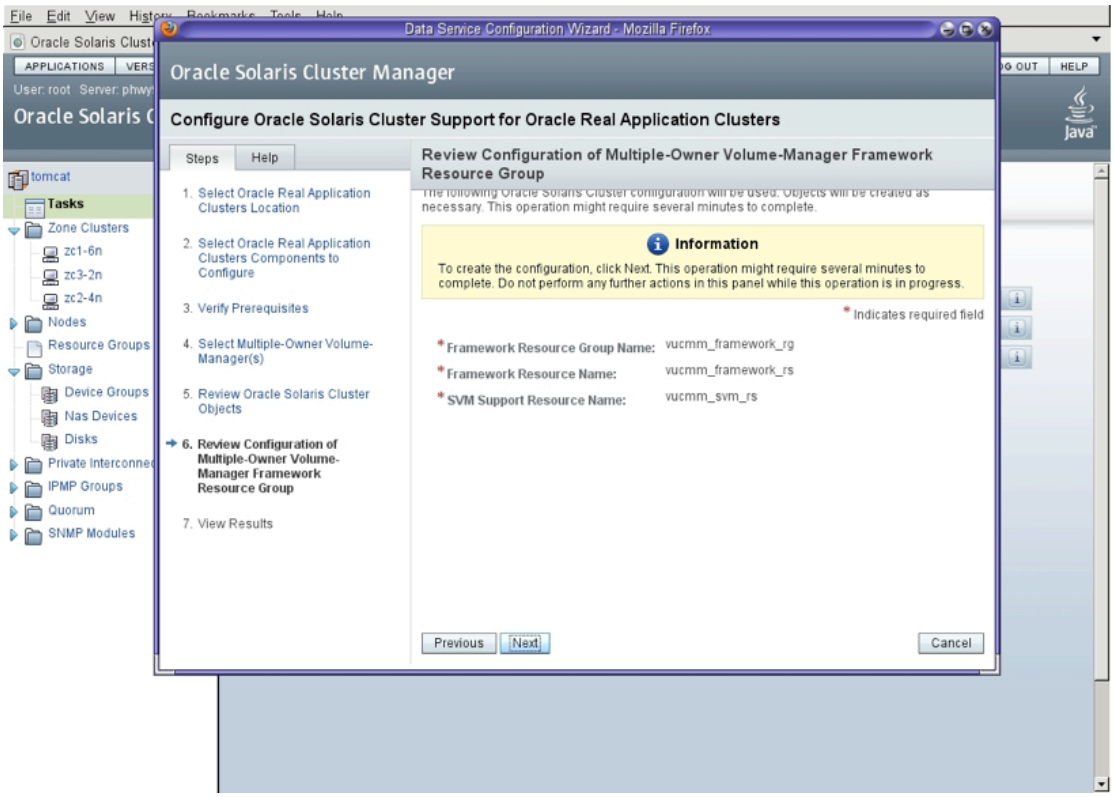

Figure 14. Screen shot — Multi-owner Volume-Manager Framework Resource Group sub-wizard.

6. Next, the Storage Resources for Oracle Files sub-wizard is executed. This sub-wizard can only be executed after the Multi-owner Volume-Manager Framework Resource Group sub-wizard completes. The administrator uses this sub-wizard to select the storage topology. This sub-wizard records information about the storage topology in a file on the node where the wizard executes.

   Options include:

   - QFS shared file system on Oracle Solaris Volume Manager with cluster feature
   - QFS shared file system on Hardware RAID
   - Oracle Solaris Volume Manager with cluster feature
   - Hardware RAID without a Volume Manager (this option is currently used when supporting Oracle RAC with ASM.)

   The Storage Resources for Oracle Files sub-wizard is used to specify storage for the following:

   - Clusterware executable files
   - Database executable files
   - Clusterware data

- Database data

There can be multiple databases in a single cluster. In this event, multiple storage locations are specified using this sub-wizard.

Figure 15 shows a screen snapshot from this sub-wizard that lists the steps that were performed. The same screen snapshot shows the sub-wizard presenting a window containing all of the commands that the system executed in response to the decisions made by the administrator in the course of working through the process of this sub-wizard.

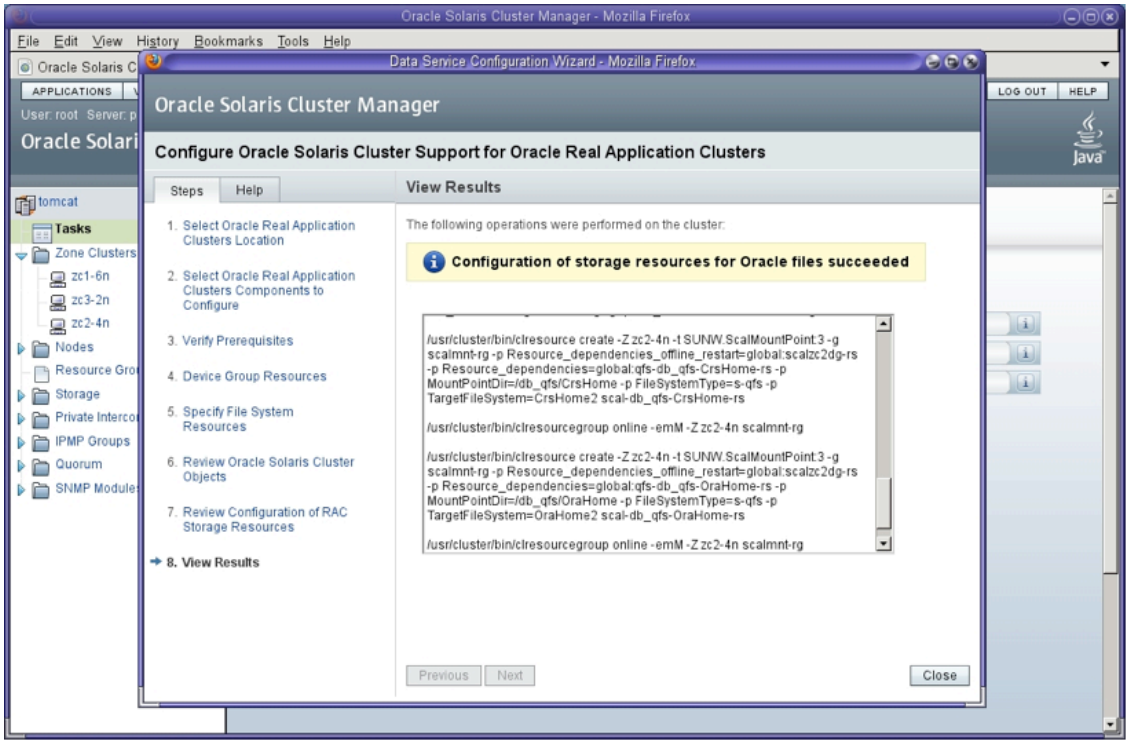

Figure 15. Screen shot — Storage Resources for Oracle Files sub-wizard.

7. At this point the administrator must leave the configuration wizard.

8. Log in to a zone cluster node and install the Oracle RAC software. The zone cluster appears as a traditional cluster to Oracle RAC. Perform the same Oracle installation and configuration steps that an administrator would do if the installation were in the global cluster. Oracle software installation is typically done from one node by invoking the Oracle Universal Installer, shipped with the Oracle Database product. The Oracle Universal Installer takes care of the installation of Clusterware and Oracle RDBMS software on all of the selected cluster nodes.

   Figure 16 shows a screen snapshot taken when executing the Oracle RAC installation software. In this example, the administrator is specifying the name of the cluster.

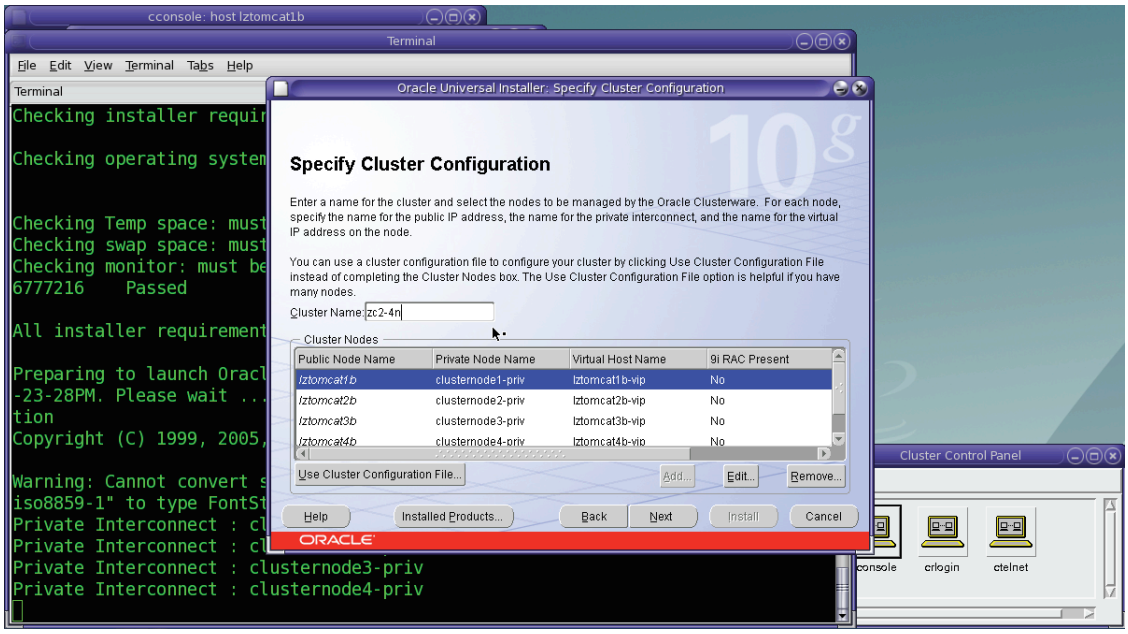

Figure 16. Screen shot — specify cluster configuration.

9. As part of the database creation the Database Configuration Assistant (`dbca`) is used, and as part of that process the user configures ASM. In this example, zone cluster zc2-4n will use ASM to manage the data. The next step creates ASM disk groups using Oracle Solaris Volume Manager volumes, as shown in Figure 17.

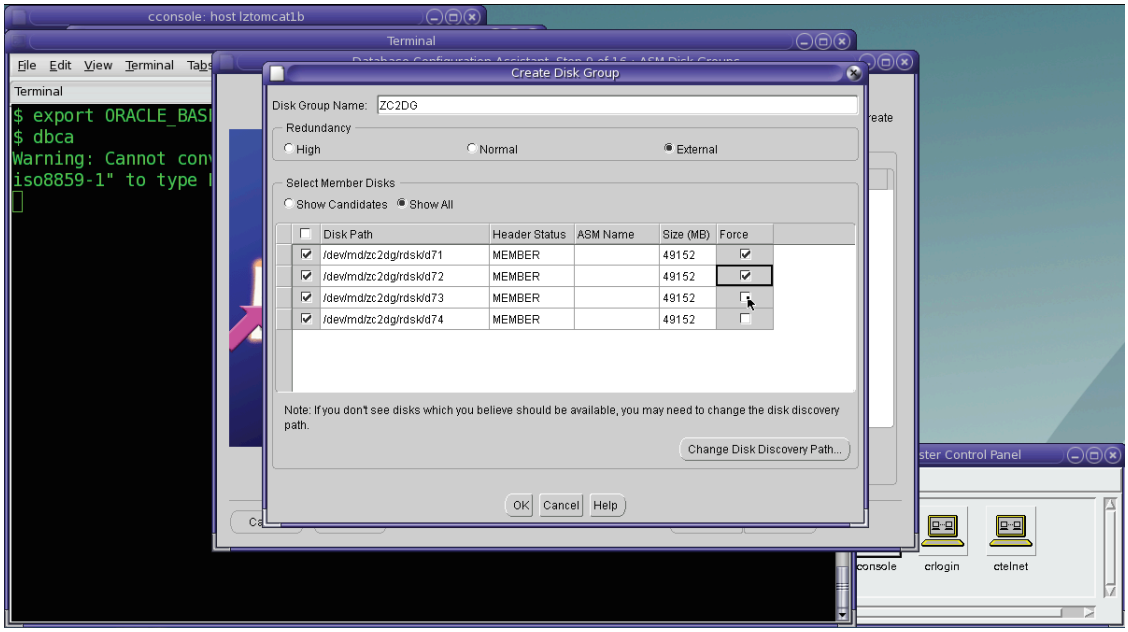

Figure 17. Screen shot — creating ASM disk group using Oracle Solaris Volume Manager devices.

10. Next, create one or more Oracle RAC database(s). Figure 18 shows a screen snapshot of one part of the database creation process.

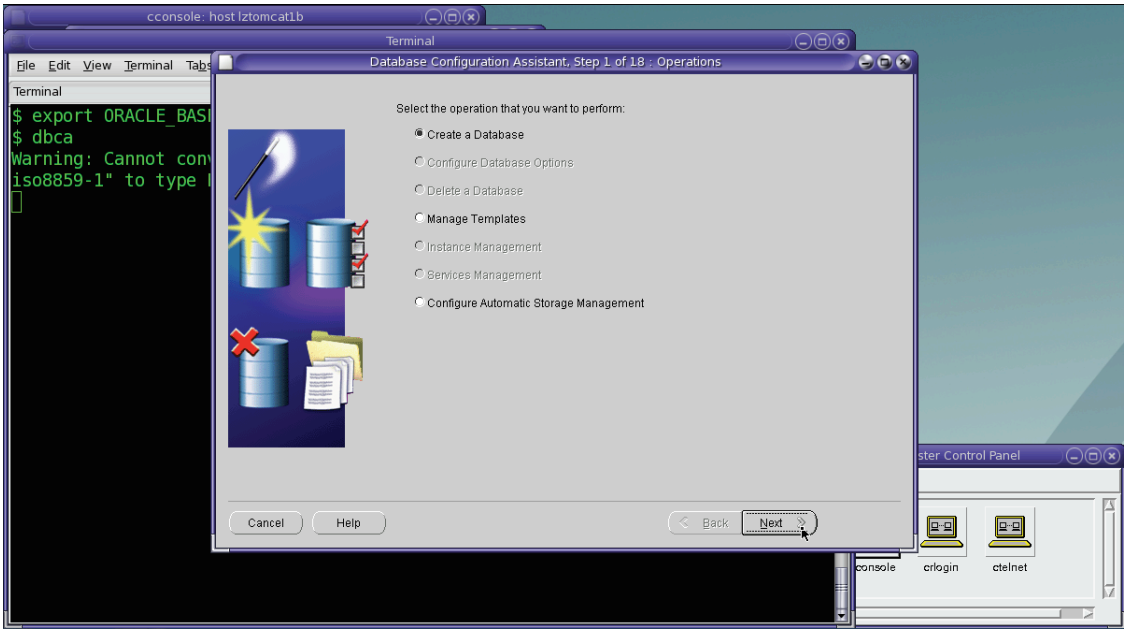

Figure 18. Screen shot — database creation.

11. At this point the administrator leaves the zone cluster and logs in to the global cluster. The Oracle Solaris Cluster Configuration wizard is a browser-based user interface. The administrator could keep one window open with the browser and use another window to log in to the zone cluster. The important point is that the administrator will now do work from the global zone.

12. Enter the Oracle Solaris Cluster Configuration Wizard for RAC, and choose the Oracle Clusterware Framework Resource sub-wizard. This sub-wizard is used to specify the Clusterware installation directory property.

13. Enter the Oracle Solaris Cluster Configuration Wizard for RAC again, and choose the Automatic Storage Management (ASM) sub-wizard. Figure 19 shows a screenshot taken during the use of the sub-wizard. This sub-wizard is used to specify the various ASM properties, including the following:

    - Oracle home directory for the ASM instances
    - The ASM instance numbers

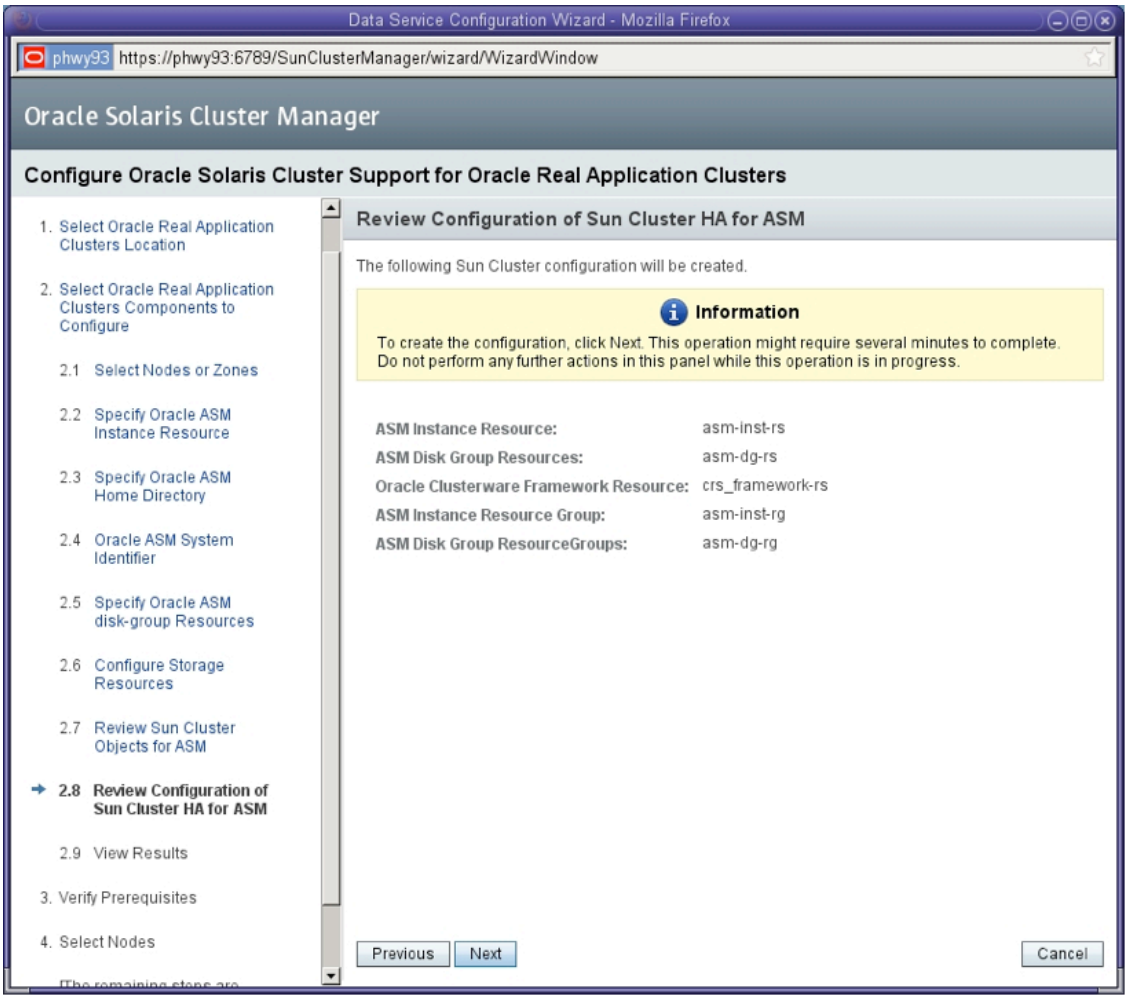

Figure 19. Screen shot — Automatic Storage Management sub-wizard

14. Enter the Oracle Solaris Cluster Configuration Wizard for RAC for the final time, and choose the Resources for Oracle Real Application Clusters Database Instances sub-wizard. This sub-wizard is used to specify the various Oracle properties, including the following:

    - Oracle RAC components (server and listener for Oracle RAC 9i only)
    - Oracle home directory
    - Clusterware home directory (RAC 10g/11g only)
    - Oracle System Identifier
    - Logical hostname (RAC 9i only)
    - Storage
    - Database name(s)

The process also includes creating resource groups, resources, resource group affinities, and resource dependencies.

For more details, see the *Oracle Solaris Cluster Data Service for Oracle RAC Guide for Oracle Solaris OS* documentation.

15. The Oracle RAC database is now ready to do work. Next, load the Order Entry schema into the Oracle RAC database.

16. Go to the client machine and start up a Swingbench Order Entry workload on the database in zone cluster zc2-4n.

    Figure 20 shows a screen snapshot of this workload.

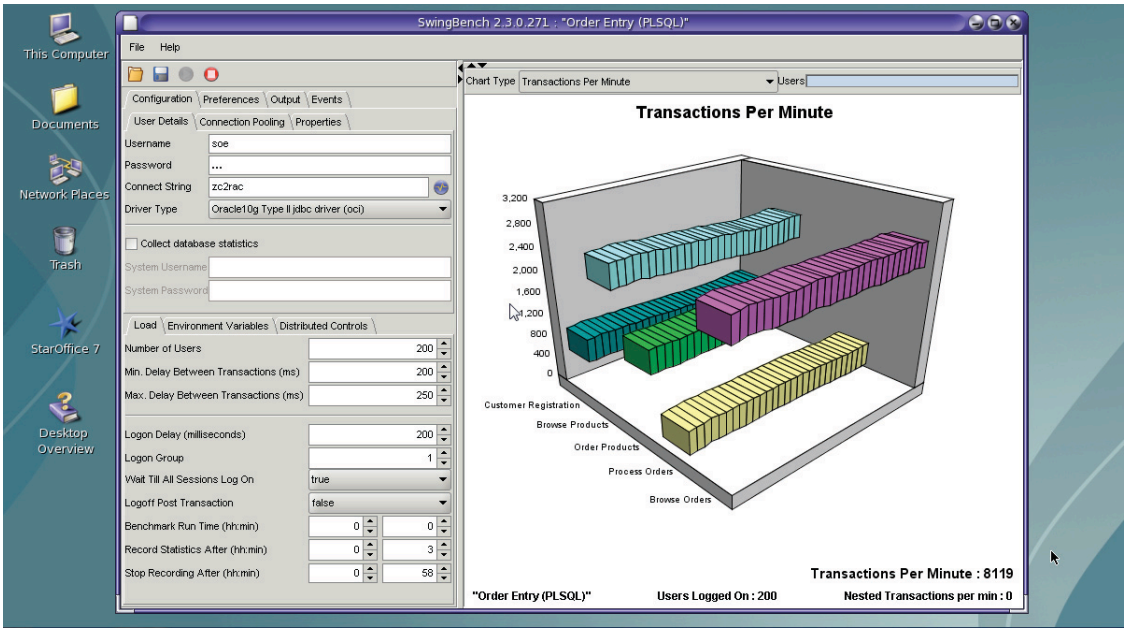

Figure 20. Screen shot — workload.

17. The Oracle Enterprise Manager in zone cluster zc2-4n can be used to observe information about that database, as shown in Figure 21.

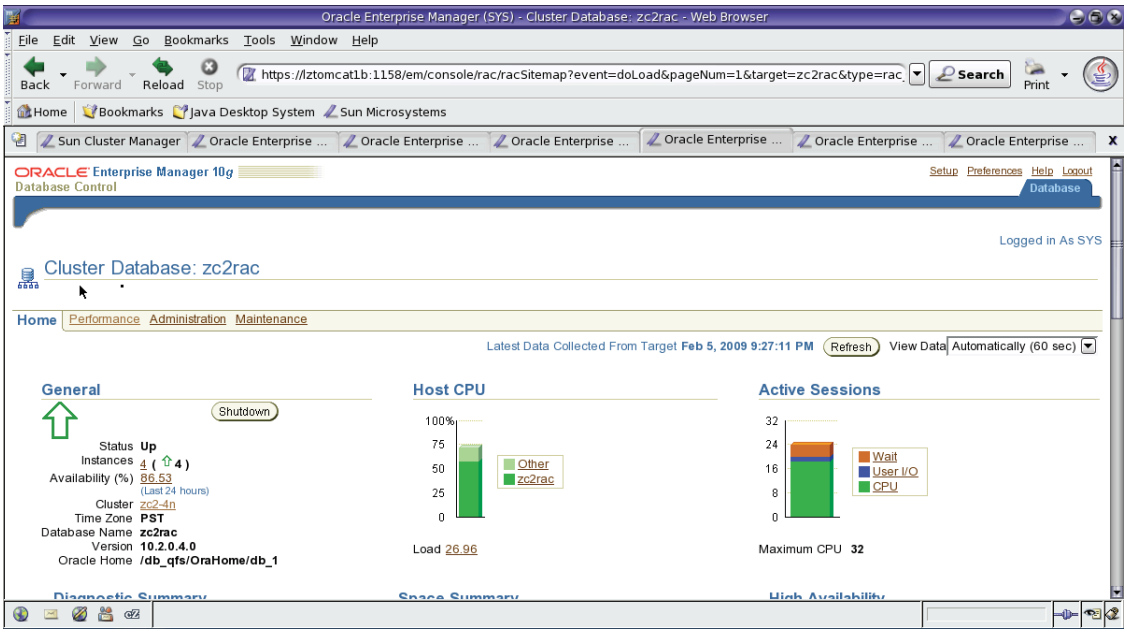

Figure 21. Screen shot — displaying information about the database.

18. A variety of status information about zone cluster zc2-4n can be observed using the Oracle Solaris Cluster Manager GUI. Figure 22 shows status information about RGM resources in zone cluster zc2-4n.

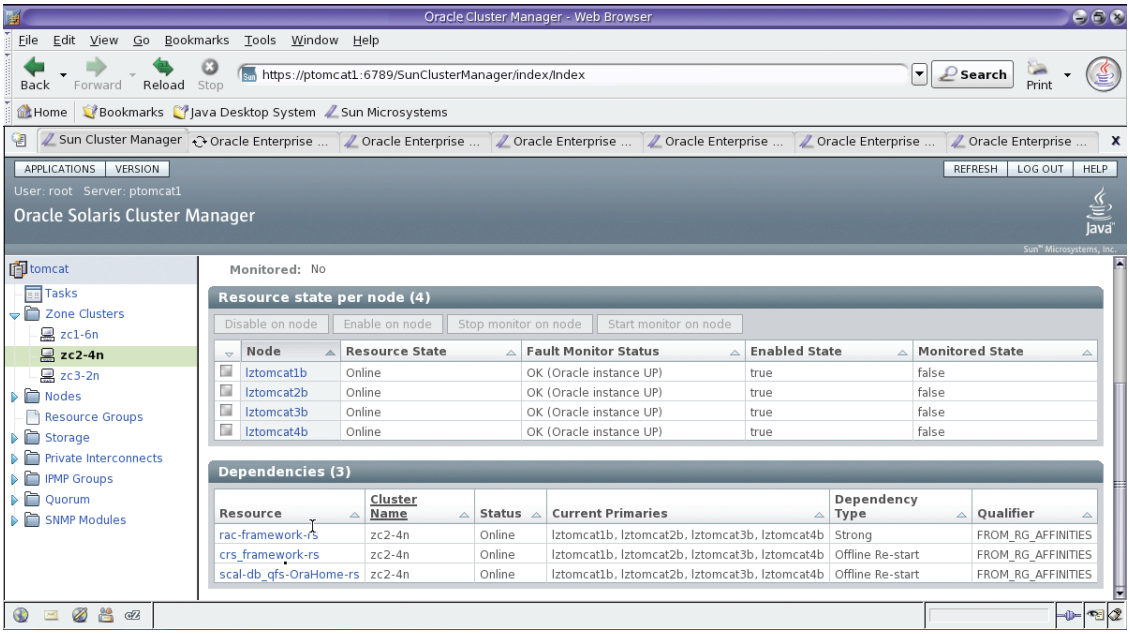

Figure 22. Screen shot — status of zone cluster zc2-4n.

19. Zone cluster zc2-4n is now ready for operations.

Oracle RAC Configurations

This section describes the various Oracle RAC configurations that zone clusters support. These configurations vary by the type of data storage, and include:

- Oracle RAC on QFS shared file systems with Oracle Solaris Volume Manager
- Oracle RAC on QFS shared file systems with hardware RAID
- Oracle RAC on Oracle Solaris Volume Manager for Oracle Solaris Cluster
- Oracle RAC on Automatic Storage Management (ASM)— supported for Oracle RAC 10g and 11g; not supported on Oracle RAC 9i because ASM was first introduced with the Oracle RAC 10g release
- Oracle RAC on NAS
- Oracle RAC on hardware RAID

Note – Different zone clusters on the same physical cluster can support different Oracle RAC configurations.

Oracle RAC software from any zone cluster node can always directly access the storage subsystem used for data. In some cases the storage subsystem has multiple layers, such as a file system and a volume manager volume. The top layer of the storage subsystem is always present in the zone cluster. For example, the file system resides in the zone cluster, while the volume manager volume resides in the global cluster. In many storage configurations, Oracle Solaris Cluster manages the availability of the storage subsystem components, such as file systems and storage devices. The storage configurations have the same resources and resource relationships on each node. For simplicity, the figures in the following sections show just the resources and their relationships on one node.

Zone clusters support RAC 9i, 10g, and 11g in almost the same storage configurations. Oracle RAC 10g and RAC 11g are supported in the same way for any particular storage configuration. Therefore, these releases are combined into a common section for each storage configuration. Oracle RAC 9i differs sufficiently from RAC 10g and RAC 11g, and thus these RAC 9i configurations are described in separate subsections.

Some of the storage topologies cannot be used for executables or are typically not used for executables. In these cases the subsection does not show how the executables are supported. The administrator can use a different storage solution to host the executables.

Readers should refer to the subsection appropriate for their configuration:

- "Oracle RAC 10g/11g Using QFS on Oracle Solaris Volume Manager"
- "Oracle RAC 10g/11g Using QFS on RAID"
- "Oracle RAC 10g/11g on Oracle Solaris Volume Manager"
- "Oracle RAC 10g/11g on ASM"
- "Oracle RAC 10g/11g on NAS"
- "Oracle RAC 9i Using QFS on Oracle Solaris Volume Manager"

- "Oracle RAC 9i Using QFS on RAID"
- "Oracle RAC 9i on Oracle Solaris Volume Manager"
- "Oracle RAC 9i on NAS"
- "Oracle RAC 9i/10g/11g on Hardware RAID"

Oracle RAC 10g/11g Using QFS on Oracle Solaris Volume Manager

Zone clusters support Oracle RAC 10g/11g using the QFS shared file system configured on top of an Oracle Solaris Volume Manager for Oracle Solaris Cluster shared disk set.

Figure 23 includes an example for this configuration that shows the zone locations and relationships of the resource groups, resources, resource dependencies, and resource group affinities that control resource group machine locations. Note that in releases prior to 11gR2, shared QFS can be used for Oracle Clusterware installation, while with 11gR2, shared QFS cannot be used for Oracle Grid Infrastructure installation. But in all releases, shared QFS can be used for for Oracle RAC RDBMS installation.

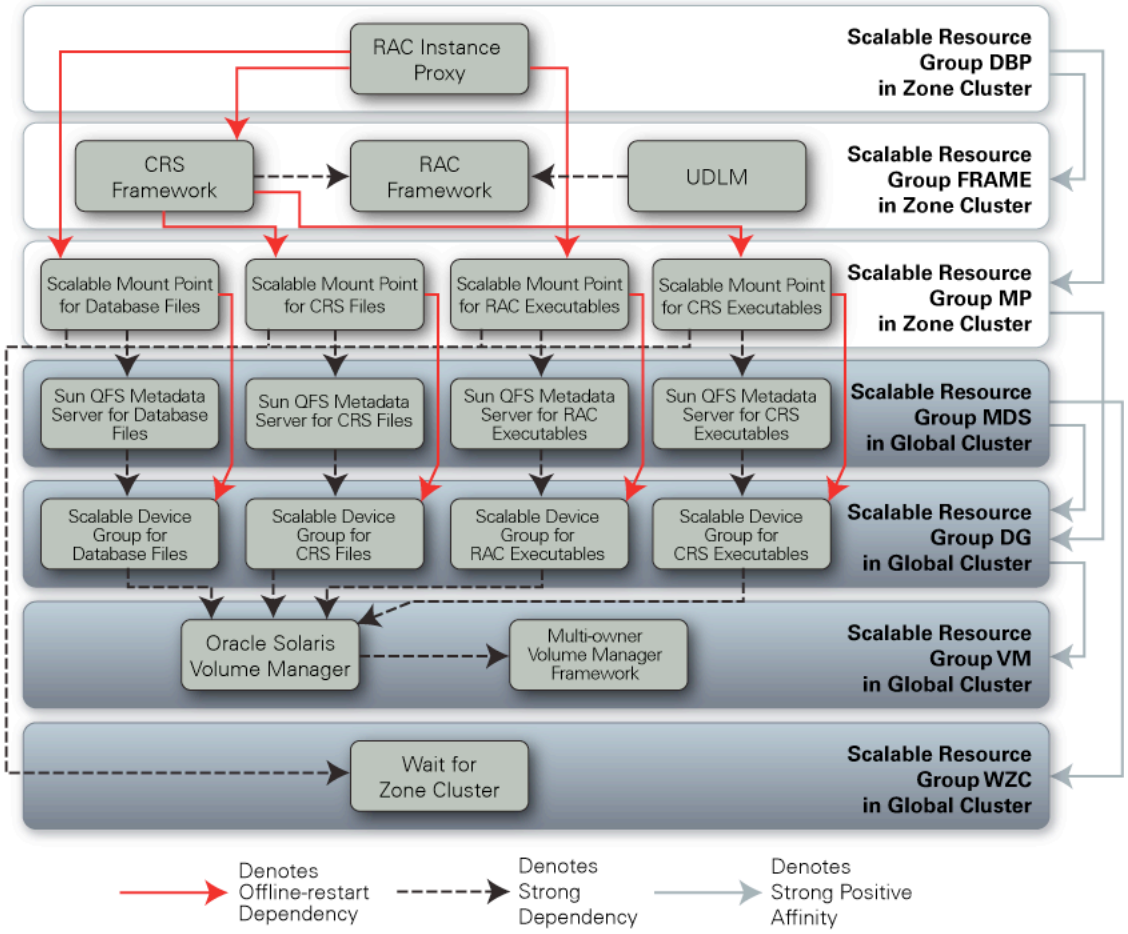

Figure 23. Resource relationships for Oracle RAC 10g/11g on shared QFS with Oracle Solaris Volume Manager.

The system administrator performs the following steps to create this configuration. Note that all steps refer to Figure 23:

1. Start in the global zone (the zone cluster does not yet exist).
2. Create the scalable resource group `VM` in the global zone. Create resources in this scalable resource group for the VUCMM framework and Oracle Solaris Volume Manager. Bring the `VM` resource group online.
3. Create the Oracle Solaris Volume Manager shared disk set(s) in exactly the same manner as if the configuration were to run in the global zone. Set the resource dependencies within the resource group `VM` as shown in Figure 23.
4. Create the scalable resource group `DG` in the global zone. Set the resource group affinities as shown in Figure 23. Create resources in this scalable resource group for scalable device group(s). The command that creates the resources also specifies the resource dependencies for that resource. Specify the resource dependencies within this resource group as shown in Figure 23. Bring the `DG` resource group online.

5. Create the QFS shared file system(s) in exactly the same manner as if the configuration were to run in the global zone. Do not mount the file system.
6. Create the zone cluster and grant access rights to the QFS shared file system(s). Install and boot the zone cluster.
7. Create the file system mount points and mount the QFS shared file system(s).
8. Create the scalable resource group WZC in the global zone with a Wait for Zone Cluster (`SUNW.wait_zc_boot`) resource for the newly created zone cluster. Bring the `WZC` resource group online. Update the resource group affinities as shown in Figure 23.
9. Create the failover resource group `MDS` in the global zone. Create QFS metadata server resource(s) in this failover resource group. Set the resource dependencies for this resource group as shown in Figure 23. Bring the `MDS` resource group online.
10. Enter the zone cluster.
11. Install the Oracle UDLM package in the zone cluster.
12. Create the scalable resource group `FRAME` in the zone cluster. Create RAC framework and UDLM resources in this scalable resource group with the resource dependencies shown in Figure 23. (The CRS Framework resource is created in a later step.) Bring the resource group FRAME online.
13. This step deals with the scalable resource group `MP`. The system validates the scalable mount point resource dependencies at creation time, and these are inter-cluster dependencies. Inter-cluster dependencies can only be specified from a global zone of a global cluster.

    Create the scalable resource group `MP` in the zone cluster. Next, enter the global zone. Set the resource group affinities for resource group `MP` as shown in Figure 23. Create scalable mount point resources in the scalable resource group `MP` belonging to the zone cluster. The command that creates the resources also specifies the resource dependencies for that resource. Specify the resource dependencies within this resource group as shown in Figure 23. Bring resource group `MP` online. Return to the zone cluster.
14. Install Oracle Clusterware and then Oracle RAC. Note that you can only install Oracle Clusterware binaries for releases prior to 11gR2 on a shared QFS file system.
15. Create the CRS Framework resource in the scalable resource group `FRAME` with the resource dependencies shown in Figure 23.
16. Create the database(s).
17. Create the proxy resources for the file system(s) in Oracle Clusterware. Then, for pre-11gR2 configurations: Update the dependency of the Clusterware resources for the database instances to include these proxy resources on a per-node basis.
18. Create the scalable resource group `DBP` in the zone cluster with the resource group affinities shown in Figure 23. Create the RAC instance proxy resource in this scalable resource group with the resource dependencies shown in Figure 23. Bring the resource group `DBP` online.

Oracle RAC 10g/11g Using QFS on RAID

Zone clusters support Oracle RAC 10g/11g using the QFS shared file system on top of hardware RAID.

Note – There is currently no resource to represent hardware RAID.

Figure 24 includes an example for this configuration that shows the zone locations and relationships of the resource groups, resources, resource dependencies, and resource group affinities that control resource group machine locations. Note that in releases prior to 11gR2, shared QFS can be used for Oracle Clusterware installation, while with 11gR2, shared QFS cannot be used for Oracle Grid Infrastructure installation. But in all releases, shared QFS can be used for for Oracle RAC RDBMS installation.

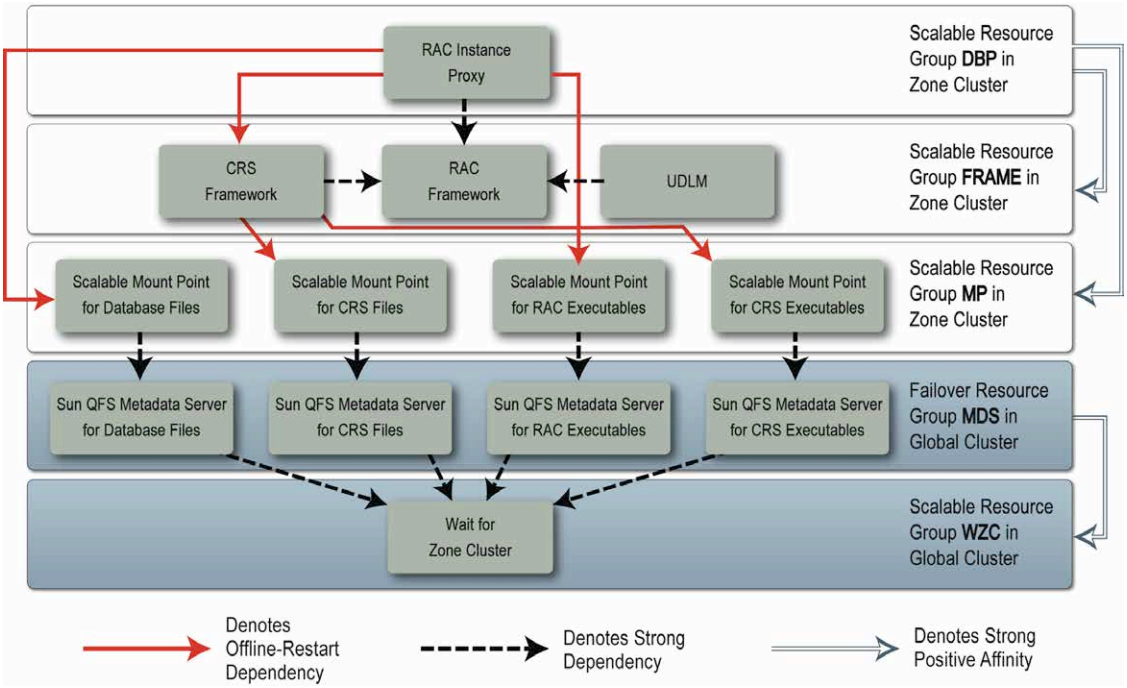

Figure 24. Resource relationships for Oracle 10g/11g on shared QFS with hardware RAID.

The system administrator performs the following steps to create this configuration. Note that all steps refer to Figure 24:

1. Start in the global zone (the zone cluster does not yet exist).
2. Create the QFS shared file system(s) in exactly the same manner as if the configuration were to run in the global zone. Do not mount the file system.
3. Create the zone cluster and grant access rights to the QFS shared file system(s). Install and boot the zone cluster.
4. Create the file system mount points and mount the QFS shared file system(s).

5. Create the scalable resource group WZC in the global zone with a Wait for Zone Cluster (`SUNW.wait_zc_boot`) resource for the newly created zone cluster. Bring the `WZC` resource group online.
6. Create the failover resource group `MDS` in the global zone. Set the resource group affinities for resource group `MP` as shown in Figure 24. Create QFS metadata server resource(s) in this failover resource group. Set the resource dependencies for this resource group as shown in Figure 24. Bring the `MDS` resource group online.
7. Enter the zone cluster.
8. Install the Oracle UDLM package in the zone cluster.
9. Create the scalable resource group `FRAME` in the zone cluster. Create RAC framework and UDLM resources in this scalable resource group with the resource dependencies shown in Figure 24. (The CRS Framework resource is created in a later step.) Bring the resource group `FRAME` online.
10. This step deals with the scalable resource group `MP`. The system validates the scalable mount point resource dependencies at creation time, and these are inter-cluster dependencies. Inter-cluster dependencies can only be specified from a global zone of a global cluster.
11. Create the scalable resource group `MP` in the zone cluster. Next, enter the global zone. Set the resource group affinities for resource group `MP` as shown in Figure 24. Create scalable mount point resources in the scalable resource group `MP` belonging to the zone cluster. The command that creates the resources also specifies the resource dependencies for that resource. Specify the resource dependencies within this resource group as shown in Figure 24. Bring resource group `MP` online. Return to the zone cluster.
12. Install Oracle Clusterware and then Oracle RAC. Note that you can only install Oracle Clusterware binaries for releases prior to 11gR2 on a shared QFS file system.
13. Create the CRS Framework resource in the scalable resource group `FRAME` with the resource dependencies shown in Figure 24.
14. Create the database(s).
15. Create the proxy resources for the file system(s) in Oracle Clusterware. Then, for pre-11gR2 configurations: Update the dependency of the Clusterware resources for the database instances to include these proxy resources on a per-node basis.
16. Create the scalable resource group `DBP` in the zone cluster with the resource group affinities shown in Figure 24. Create the RAC instance proxy resource in this scalable resource group with the resource dependencies shown in Figure 24. Bring the resource group `DBP` online.

Oracle RAC 10g/11g on Oracle Solaris Volume Manager

Zone clusters support Oracle RAC 10g/11g using the Oracle Solaris Volume Manager for Oracle Solaris Cluster. In this configuration, the disk sets are directly accessible to the zone. This is supported,

but the administrator should be aware that programs within the zone can now issue `ioctl` function calls directly to the device.

Figure 25 includes an example for this configuration that shows the zone locations and relationships of the resource groups, resources, resource dependencies, and resource group affinities that control resource group machine locations.

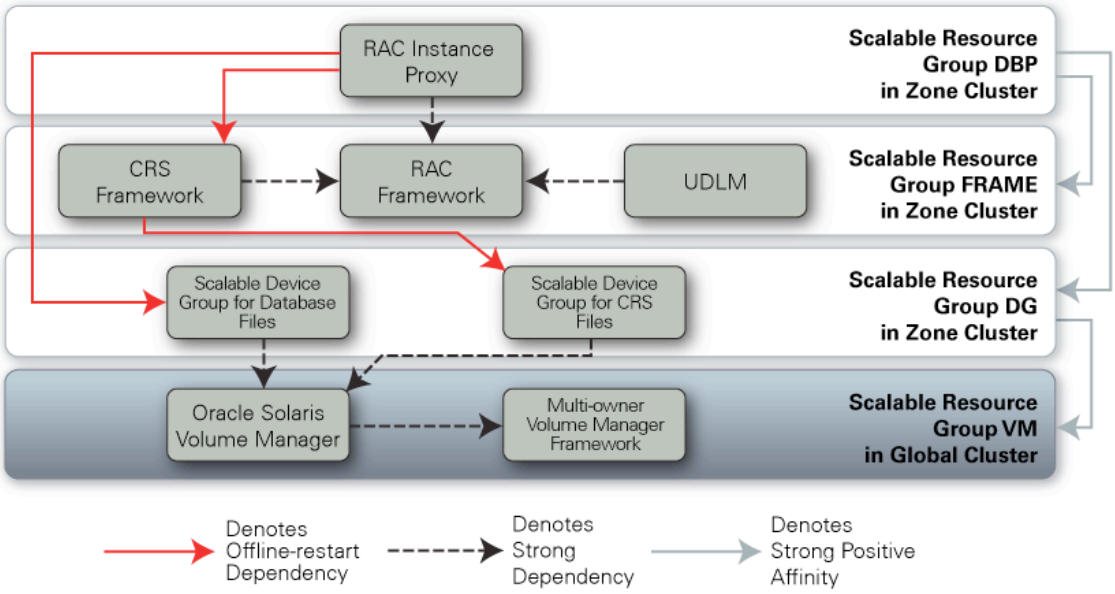

Figure 25. Resource relationships for Oracle RAC 10g/11g on Oracle Solaris Volume Manager.

The system administrator performs the following steps to create this configuration. Note that all steps refer to Figure 25:

1. Start in the global zone (the zone cluster does not yet exist).
2. Create the scalable resource group `VM` in the global zone. Create resources in this scalable resource group for VUCMM framework and Oracle Solaris Volume Manager. Bring the `VM` resource group online.
3. Create the Oracle Solaris Volume Manager shared disk set(s) in exactly the same manner as if the configuration were to run in the global zone.
4. Create the zone cluster and grant access rights to the Oracle Solaris Volume Manager disk set(s). Install and boot the zone cluster.
5. Enter the zone cluster.
6. Install the Oracle UDLM package in the zone cluster.

7. Create the scalable resource group `FRAME` in the zone cluster. Create RAC framework and UDLM resources in this scalable resource group with the resource dependencies shown in Figure 25. (The CRS Framework resource is created in a later step.) Bring the resource group `FRAME` online.
8. Create the scalable resource group `DG` in the zone cluster. Set the resource group affinities as shown in Figure 25. Create resources in this scalable resource group for scalable device group(s). The command that creates the resource also specifies the resource dependencies for that resource. Specify the resource dependencies within this resource group as shown in Figure 25. Bring the `DG` resource group online.
9. Install Oracle Clusterware and then Oracle RAC.
10. Create the CRS Framework resource in the scalable resource group `FRAME` with the resource dependencies shown in Figure 25.
11. Create the database(s).
12. Create the proxy resources for the devices(s) in Oracle Clusterware. Then, for pre-11gR2 configurations: Update the dependency of the Clusterware resources for the database instances to include these proxy resources on a per-node basis.
13. Create the scalable resource group `DBP` in the zone cluster with the resource group affinities shown in Figure 25. Create the RAC instance proxy resource in this scalable resource group with the resource dependencies shown in Figure 25. Bring the resource group `DBP` online.

Oracle RAC 10g/11g on ASM using Hardware RAID

Zone clusters support Oracle RAC 10g/11g using Oracle Automatic Storage Management (ASM) deployed on hardware RAID.

In this configuration, the disks are directly accessible to the zone. ASM works directly with the DID disks. This is supported, but the administrator should be aware that programs within the zone can now issue `ioctl` function calls directly to the device.

Figure 26 includes an example for this configuration that shows the zone locations and relationships of the resource groups, resources, resource dependencies, and resource group affinities that control resource group machine locations.

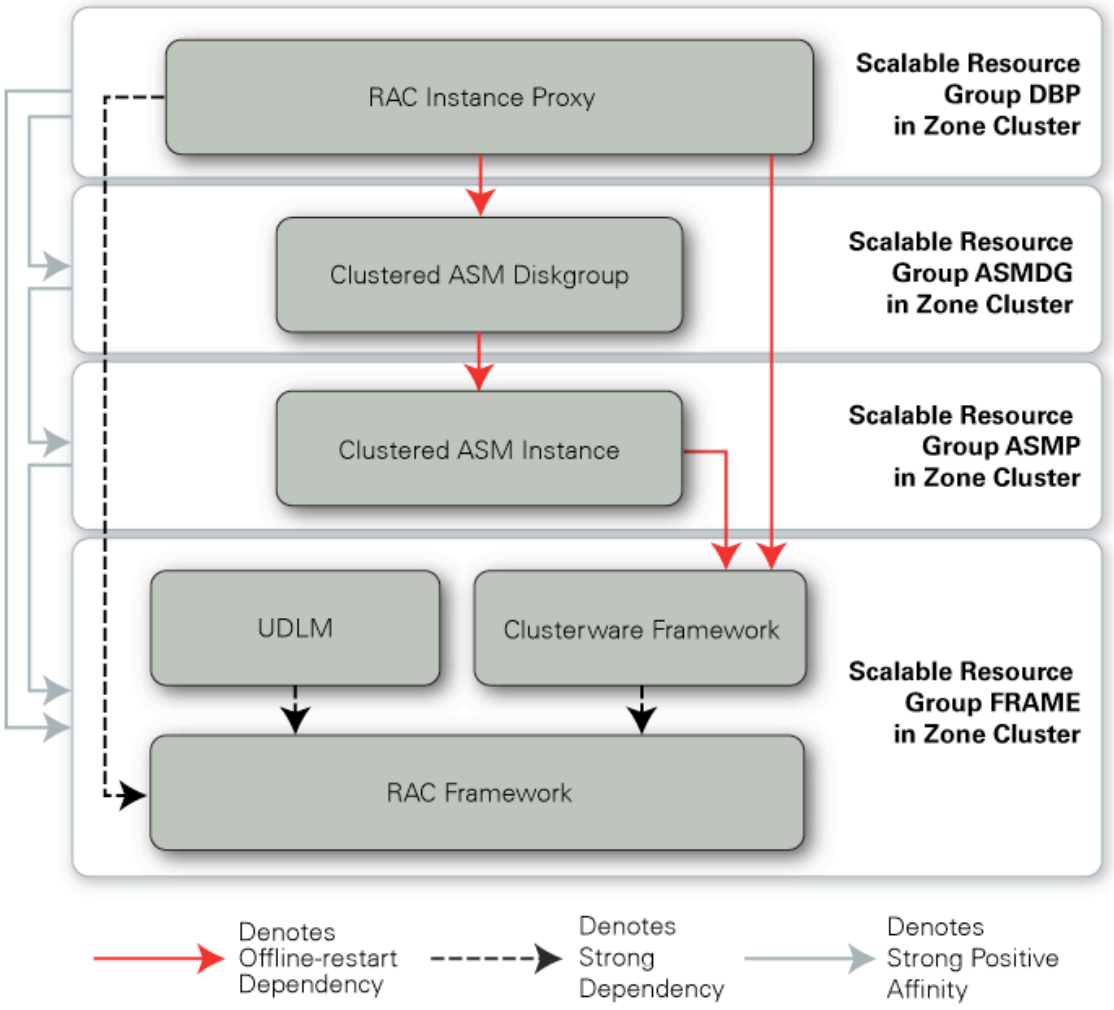

Figure 26. Resource relationships for Oracle RAC 10g/11g on ASM using hardware RAID.

The system administrator performs the following steps to create this configuration. Note that all steps refer to Figure 26:

1. Start in the global zone (the zone cluster does not yet exist).
2. Create the zone cluster and grant access rights to the disk(s). Install and boot the zone cluster.
3. Enter the zone cluster.
4. Install the Oracle UDLM package in the zone cluster.
5. Create the scalable resource group `FRAME` in the zone cluster. Create RAC framework and UDLM resources in this scalable resource group with the resource dependencies shown in Figure 26. Bring the resource group `FRAME` online.
6. Install Oracle Clusterware. If 11g is used, configure ASM to work with the storage devices configured into the zone cluster at this point.

7. Create the CRS Framework resource in the scalable resource group `FRAME` with the resource dependencies shown in Figure 26.
8. Install Oracle RAC. If 10g is used, configure ASM to work with the storage devices configured into the zone cluster at this point.
9. Create the scalable resource group `ASMP` in the zone cluster. Set the resource group affinities as shown in Figure 26. Create clustered ASM instance resource in this scalable resource group with the resource dependencies shown in Figure 26.
10. Create the scalable resource group `ASMDG` in the zone cluster. Set the resource group affinities as shown in Figure 26. Create clustered ASM disk group resource in this scalable resource group with the resource dependencies shown in Figure 26. If 11gR2 is used, then configure a `SUNW.scalable_asm_diskgroup_proxy` resource, rather than a `SUNW.asm_diskgroup resource`.
11. Create the database(s).
12. Create the scalable resource group `DBP` in the zone cluster. Set the resource group affinities as shown in Figure 26. Create the RAC instance proxy resource in this scalable resource group with the resource dependencies shown in Figure 26. Bring the resource group `DBP` online.

Oracle RAC 10g/11g on ASM using Solaris Volume Manager

Zone clusters support Oracle RAC 10g/11g using Oracle Automatic Storage Management (ASM) deployed on Oracle Solaris Volume Manager for Oracle Solaris Cluster volumes.

In this configuration, the disks are directly accessible to the zone. ASM works directly with Oracle Solaris Volume Manager for Oracle Solaris Cluster volumes. This is supported, but the administrator should be aware that programs within the zone can now issue `ioctl` function calls directly to the device.

Figure 27 includes an example for this configuration that shows the zone locations and relationships of the resource groups, resources, resource dependencies, and resource group affinities that control resource group machine locations.

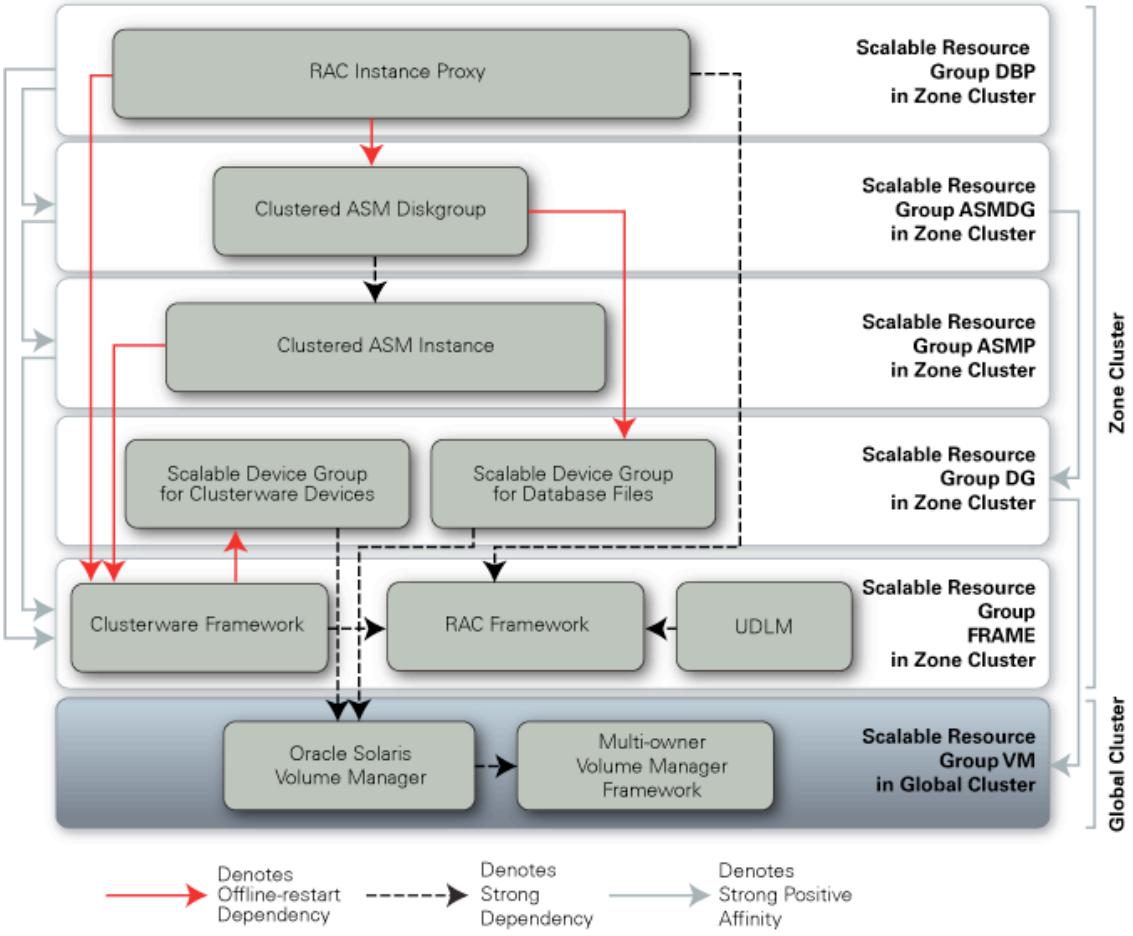

Figure 27. Resource relationships for Oracle RAC 10g/11g on ASM using Solaris Volume Manager.

The system administrator performs the following steps to create this configuration. Note that all steps refer to Figure 27:

1. Start in the global zone (the zone cluster does not yet exist).
2. Create the scalable resource group `VM` in the global zone. Create resources in this scalable resource group for VUCMM framework and Oracle Solaris Volume Manager. Bring the `VM` resource group online.
3. Create the Oracle Solaris Volume Manager shared disk set(s) in exactly the same manner as if the configuration were to run in the global zone.
4. Create the zone cluster and grant access rights to the Oracle Solaris Volume Manager disk set(s). Install and boot the zone cluster.
5. Enter the zone cluster.
6. Install the Oracle UDLM package in the zone cluster.

7. Create the scalable resource group `FRAME` in the zone cluster. Create RAC framework and UDLM resources in this scalable resource group with the resource dependencies shown in Figure 27. (The CRS Framework resource is created in a later step.) Bring the resource group `FRAME` online.
8. Create the scalable resource group `DG` in the zone cluster. Set the resource group affinities as shown in Figure 27. Create resources in this scalable resource group for scalable device group(s). The command that creates the resource also specifies the resource dependencies for that resource. Specify the resource dependencies within this resource group as shown in Figure 27. If 11g is used and the voting disk and OCR file are held in ASM, then only the device group resource for ASM is required. Bring the `DG` resource group online.
9. Install Oracle Clusterware. If 11g is used, configure ASM to work with the storage devices configured into the zone cluster at this point.
10. Create the CRS Framework resource in the scalable resource group `FRAME` with the resource dependencies shown in Figure 27.
11. Install Oracle RAC. If 10g is used, configure ASM to work with the storage devices configured into the zone cluster at this point.
12. Create the scalable resource group `ASMP` in the zone cluster. Set the resource group affinities as shown in Figure 27. Create clustered ASM instance resource in this scalable resource group with the resource dependencies shown in Figure 27.
13. Create the scalable resource group `ASMDG` in the zone cluster. Set the resource group affinities as shown in Figure 27. Create clustered ASM disk group resource in this scalable resource group with the resource dependencies shown in Figure 27. If 11gR2 is used, then configure a `SUNW.scalable_asm_diskgroup_proxy` resource, rather than a `SUNW.asm_diskgroup resource`.
14. Create the database(s).
15. Create the scalable resource group `DBP` in the zone cluster. Set the resource group affinities as shown in Figure 27. Create the RAC instance proxy resource in this scalable resource group with the resource dependencies shown in Figure 27. Bring the resource group `DBP` online.

Oracle RAC 10g/11g on NAS

Zone clusters support Oracle RAC 10g/11g using an NFS mounted file system from a Network Attached Storage (NAS) device. The NFS mounts happen inside the zone cluster.

The scalable mount point resource (`SUNW.ScalMountPoint`) monitors the status of the NFS file system from the NAS device, and this resource resides within the zone cluster.

Figure 28 includes an example for this configuration that shows the zone locations and relationships of the resource groups, resources, resource dependencies, and resource group affinities that control resource group machine locations.

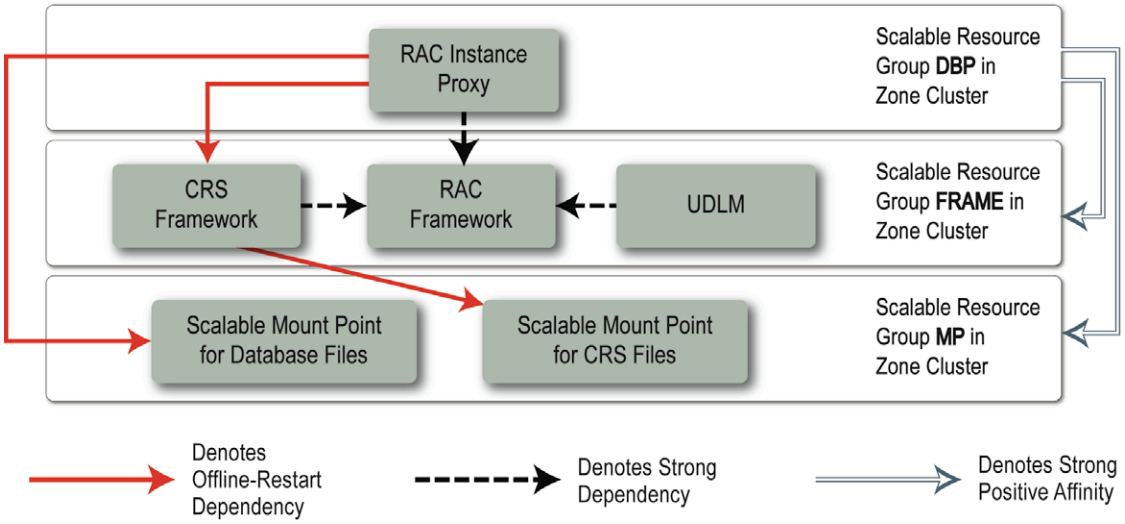

Figure 28. Resource relationships for Oracle RAC 10g/11g on NAS.

The system administrator performs the following steps to create this configuration. Note that all of the steps refer to Figure 28:

1. Start in the global zone (the zone cluster does not yet exist).
2. Create the zone cluster. Install and boot the zone cluster.
3. Enter the zone cluster.
4. Create the NFS client mounts for the NFS file systems on the NAS device.
5. Install the Oracle UDLM package in the zone cluster.
6. Create the scalable resource group `FRAME` in the zone cluster. Create RAC framework and UDLM resources in this scalable resource group with the resource dependencies shown in Figure 28. (The Clusterware framework resource is created in a later step.) Bring the resource group `FRAME` online.
7. Create the scalable resource group `MP` in the zone cluster. Create scalable mount point resources in this scalable resource group. From the global zone set the resource dependencies and resource group affinities as shown in Figure 28. Bring resource group `MP` online.
8. Install Oracle Clusterware and then Oracle RAC.
9. Create the Clusterware Framework resource in the scalable resource group `FRAME` with the resource dependencies shown in Figure 28.
10. Create the database(s).

11. Create the proxy resources for the file system(s) in Oracle Clusterware. Then, for pre-11gR2 configurations: Update the dependency of the Clusterware resources for the database instances to include these proxy resources on a per-node basis.
12. Create the scalable resource group `DBP` in the zone cluster. Set the resource group affinities as shown in Figure 28. Create the RAC instance proxy resource in this scalable resource group with the resource dependencies shown in Figure 28. Bring the resource group `DBP` online.

Oracle RAC 9i Using QFS on Oracle Solaris Volume Manager

Zone clusters support Oracle RAC 9i using the QFS shared file system on top of an Oracle Solaris Volume Manager for Oracle Solaris Cluster shared disk set.

Figure 29 includes an example for this configuration that shows the zone locations and relationships of the resource groups, resources, resource dependencies, and resource group affinities that control resource group machine locations. Note that this example shows the RAC executables also placed in a QFS shared file system.

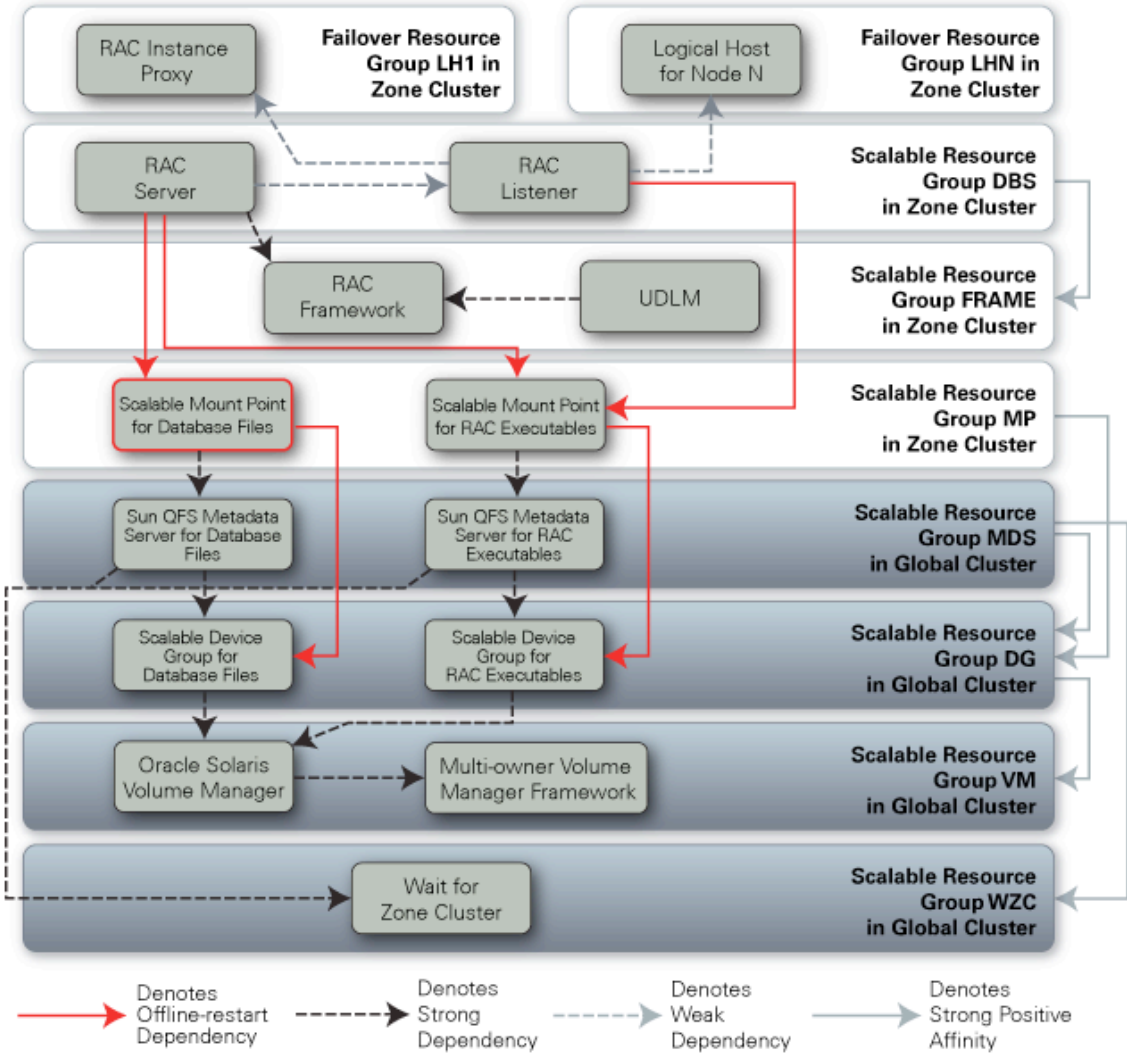

Figure 29. Resource relationships for Oracle RAC 9i on shared QFS with Oracle Solaris Volume Manager.

The system administrator performs the following steps to create this configuration. Note that all of the steps refer to Figure 29:

1. Start in the global zone (the zone cluster does not yet exist).
2. Create the scalable resource group `VM` in the global zone. Create resources in this scalable resource group for RAC framework and Oracle Solaris Volume Manager. Bring the `VM` resource group online.
3. Create the Oracle Solaris Volume Manager shared disk set(s) in exactly the same manner as if the configuration were to run in the global zone. Set the resource dependencies within this resource group as shown in Figure 29.

4. Create the scalable resource group `DG` in the global zone. Set the resource group affinities as shown in Figure 29. Create resources in this scalable resource group for scalable device group(s). The command that creates the resources also specifies the resource dependencies for that resource. Specify the resource dependencies within this resource group as shown in Figure 29. Bring the `DG` resource group online.
5. Create the QFS shared file system(s) in exactly the same manner as if the configuration were to run in the global zone. Do not mount the file system.
6. Create the zone cluster and grant access rights to the QFS shared file system(s) and network resources for logical hosts. Install and boot the zone cluster.
7. Create the file system mount points and mount the QFS shared file system(s).
8. Create the scalable resource group `WZC` in the global zone with a Wait for Zone Cluster (`SUNW.wait_zc_boot`) resource for the newly created zone cluster. Bring the `WZC` resource group online.
9. Create the failover resource group `MDS` in the global zone. Set the resource group affinities as shown in Figure 29. Create QFS metadata server resource(s) in this failover resource group. Set the resource dependencies for this resource group as shown in Figure 29. Bring the `MDS` resource group online.
10. Enter the zone cluster.
11. Install the Oracle UDLM package in the zone cluster.
12. Create one failover resource group per node (`LH1`, `LH2`, and so on) and create a logical host resource within that failover resource group. Bring each of these resource groups online.
13. Create the scalable resource group `FRAME` in the zone cluster. Create RAC framework and UDLM resources in this scalable resource group with the resource dependencies shown in Figure 29. Bring the resource group `FRAME` online.
14. This step deals with the scalable resource group `MP`. The system validates the scalable mount point resource dependencies at creation time, and these are inter-cluster dependencies. Inter-cluster dependencies can only be specified from a global zone of a global cluster.

    Create the scalable resource group `MP` in the zone cluster. Next, enter the global zone. Set the resource group affinities for resource group `MP` as shown in Figure 29. Create scalable mount point resources in the scalable resource group `MP` belonging to the zone cluster. The command that creates the resources also specifies the resource dependencies for that resource. Specify the resource dependencies within this resource group as shown in Figure 29. Bring resource group `MP` online. Return to the zone cluster.
15. Install Oracle RAC.
16. Create the database(s).

17. Create the scalable resource group `DBS` in the zone cluster. Set the resource group affinities as shown in Figure 29. Create the RAC server and RAC listener resources in this scalable resource group with the resource dependencies shown in Figure 29. Bring the resource group `DBS` online.

Oracle RAC 9i Using QFS on RAID

Zone clusters support Oracle RAC 9i using the QFS shared file system on top of hardware RAID.

Note – There is currently no resource to represent hardware RAID.

Figure 30 includes an example for this configuration that shows the zone locations and relationships of the resource groups, resources, resource dependencies, and resource group affinities that control resource group machine locations. Note that this example shows the Oracle RAC executables also placed in a QFS shared file system.

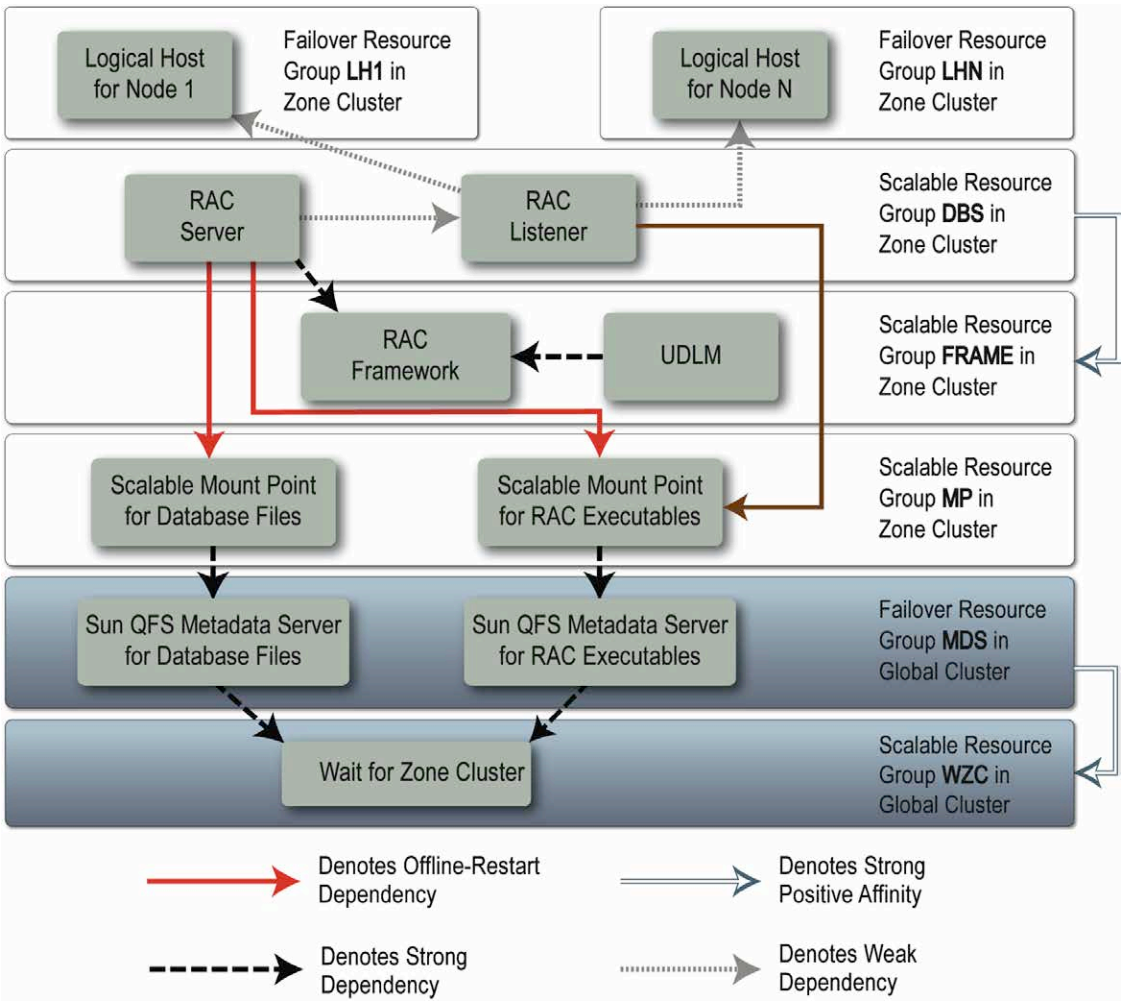

Figure 30. Resource relationships for Oracle RAC 9i on QFS shared file system with hardware RAID.

The system administrator performs the following steps to create this configuration. Note that all steps refer to Figure 30:

1. Start in the global zone (the zone cluster does not yet exist).
2. Create the QFS shared file system(s) in exactly the same manner as if the configuration were to run in the global zone. Do not mount the file system.
3. Create the zone cluster and grant access rights to the QFS shared file system(s). Install and boot the zone cluster.
4. Create the file system mount points and mount the QFS shared file system(s).
5. Create the scalable resource group `WZC` in the global zone with a Wait for Zone Cluster (`SUNW.wait_zc_boot`) resource for the newly created zone cluster. Bring the `WZC` resource group online.
6. Create the failover resource group `MDS` in the global zone. Set the resource group affinities as shown in Figure 30. Create QFS metadata server resource(s) in this failover resource group. Set the resource dependencies for this resource group as shown in Figure 30. Bring the `MDS` resource group online.
7. Enter the zone cluster.
8. Install the Oracle UDLM package in the zone cluster.
9. Create one failover resource group per node (`LH1`, `LH2`, and so on) and create a logical host resource within that failover resource group. Bring each of these resource groups online.
10. Create the scalable resource group `FRAME` in the zone cluster. Create RAC framework and UDLM resources in this scalable resource group with the resource dependencies shown in Figure 30. Bring the resource group `FRAME` online.
11. This step deals with the scalable resource group `MP`. The system validates the scalable mount point resource dependencies at creation time, and these are inter-cluster dependencies. Inter-cluster dependencies can only be specified from a global zone of a global cluster.
12. Create the scalable resource group `MP` in the zone cluster. Next, enter the global zone. Set the resource group affinities for resource group `MP` as shown in Figure 30. Create scalable mount point resources in the scalable resource group `MP` belonging to the zone cluster. The command that creates the resources also specifies the resource dependencies for that resource. Specify the resource dependencies within this resource group as shown in Figure 30. Bring resource group `MP` online. Return to the zone cluster.
13. Install Oracle RAC.
14. Create the database(s).
15. Create the scalable resource group `DBS` in the zone cluster. Set the resource group affinities as shown in Figure 30. Create the RAC server and RAC listener resources in this scalable resource group with the resource dependencies shown in Figure 30. Bring resource group `DBS` online.

Oracle RAC 9i on Oracle Solaris Volume Manager

Zone clusters support Oracle RAC 9i using the Oracle Solaris Volume Manager for Oracle Solaris Cluster.

In this configuration, the disk sets are directly accessible to the zone. This is supported, but the administrator should be aware that programs within the zone can now issue `ioctl` operations directly to the device.

Figure 31 includes an example for this configuration that shows the zone locations and relationships of the resource groups, resources, resource dependencies, and resource group affinities that control resource group machine locations.

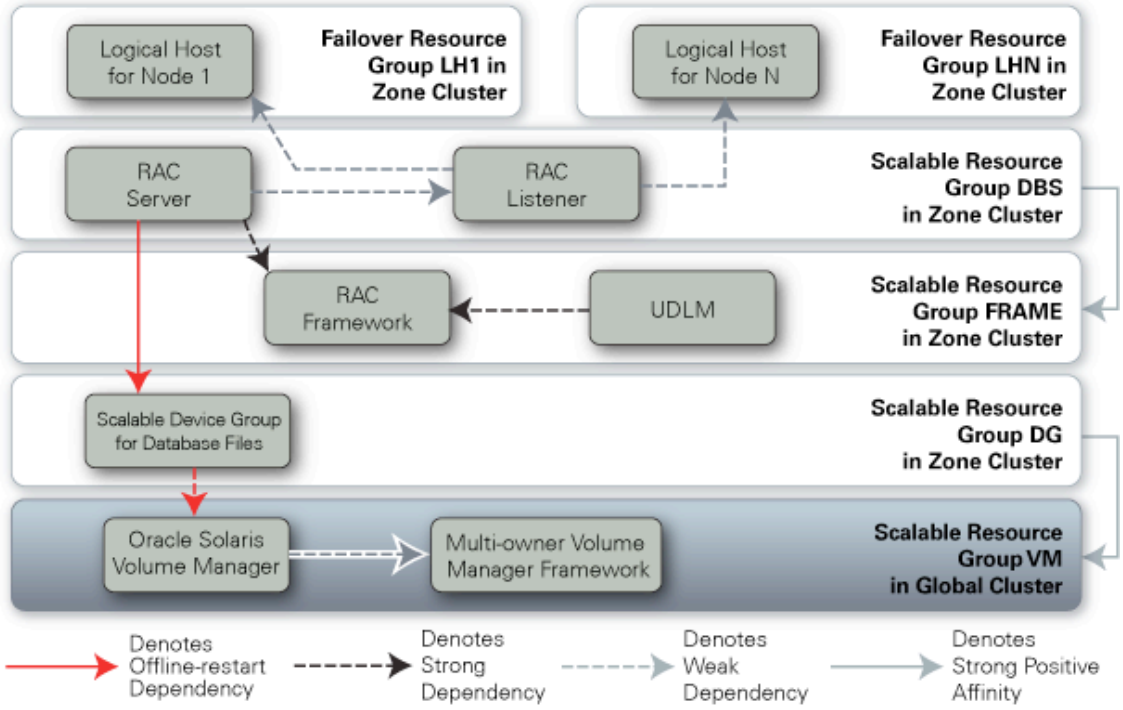

Figure 31. Resource relationships for Oracle RAC 9i on Oracle Solaris Volume Manager.

The system administrator performs the following steps to create this configuration. Note that all steps refer to Figure 31:

1. Start in the global zone (the zone cluster does not yet exist).
2. Create the scalable resource group `VM` in the global zone. Create resources in this scalable resource group for RAC framework and Oracle Solaris Volume Manager. Bring the `VM` resource group online.
3. Create the Oracle Solaris Volume Manager shared disk set(s) in exactly the same manner as if the configuration were to run in the global zone.

4. Create the zone cluster and grant access rights to the Oracle Solaris Volume Manager disk set(s). Install and boot the zone cluster.
5. Enter the zone cluster.
6. Install the Oracle UDLM package in the zone cluster.
7. Create one failover resource group per node (`LH1`, `LH2`, and so on) and create a logical host resource within that failover resource group. Bring each of these resource groups online.
8. Create the scalable resource group `FRAME` in the zone cluster. Create RAC framework and UDLM resources in this scalable resource group with the resource dependencies shown in the figure. Bring the resource group `FRAME` online.
9. Create the scalable resource group `DG` in the zone cluster. Set the resource group affinities as shown in Figure 31. Create resources in this scalable resource group for scalable device group(s). The command that creates the resource also specifies the resource dependencies for that resource. Specify the resource dependencies within this resource group as shown in Figure 31. Bring the `DG` resource group online.
10. Install Oracle RAC.
11. Create the database(s).
12. Create the scalable resource group `DBS` in the zone cluster. Set the resource group affinities as shown in Figure 31. Create the RAC server and RAC listener resources in this scalable resource group with the resource dependencies shown in Figure 31. Bring resource group `DBS` online.

Oracle RAC 9i on NAS

Zone clusters plan support for RAC 9i using an NFS mounted file system from a Network Attached Storage (NAS) device. The NFS mounts happen inside the zone cluster.

The scalable mount point resource (`SUNW.ScalMountPoint`) monitors the status of the NFS file system from the NAS device, and this resource resides within the zone cluster.

Figure 32 includes an example for this configuration that shows the zone locations and relationships of the resource groups, resources, resource dependencies, and resource group affinities that control resource group machine locations.

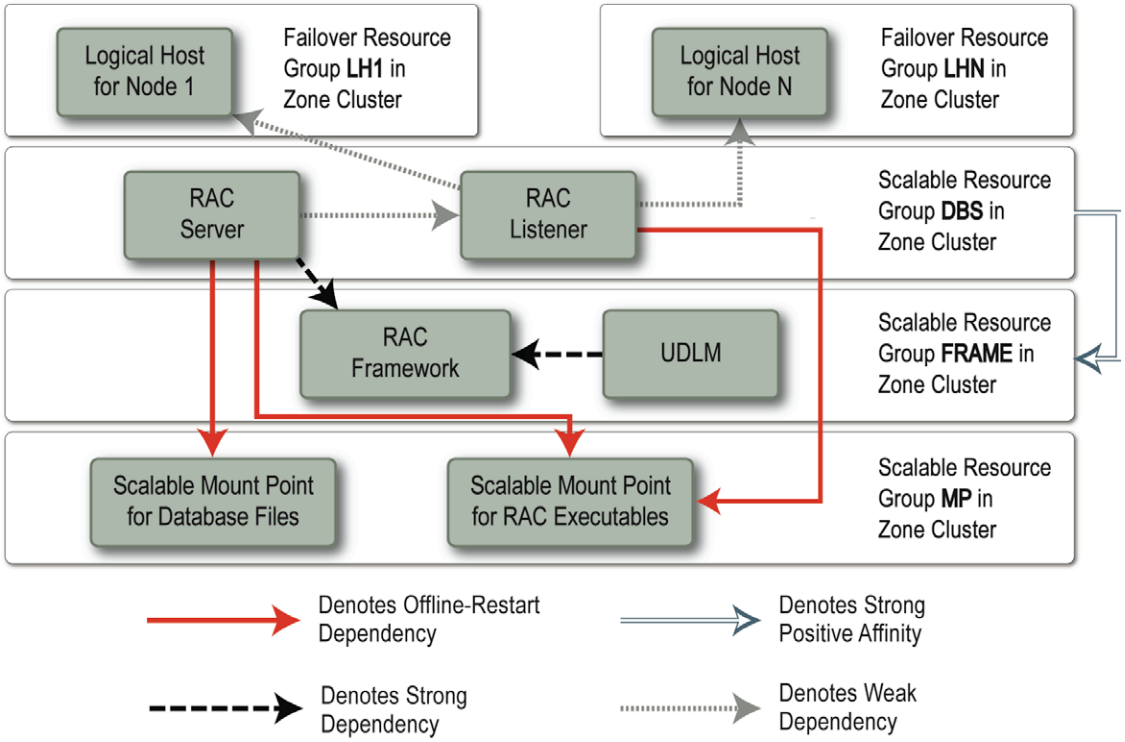

Figure 32. Resource relationships for Oracle 9i on NAS.

The system administrator performs the following steps to create this configuration. Note that all of the steps refer to Figure 32:

1. Start in the global zone (the zone cluster does not yet exist).
2. Create the zone cluster. Install and boot the zone cluster.
3. Enter the zone cluster.
4. Create the NFS client mounts for the NFS file systems on the NAS device.
5. Install the Oracle UDLM package in the zone cluster.
6. Create one failover resource group per node (`LH1`, `LH2`, and so on) and create a logical host resource within that failover resource group. Bring each of these resource groups online.
7. Create the scalable resource group `FRAME` in the zone cluster. Create RAC framework and UDLM resources in this scalable resource group with the resource dependencies shown in Figure 32. Bring the resource group `FRAME` online.
8. Create the scalable resource group `MP` in the zone cluster. Create scalable mount point resources in this scalable resource group with the resource dependencies shown in Figure 32. Bring resource group `MP` online.
9. Install Oracle RAC.
10. Create the database(s).

11. Create the scalable resource group `DBS` in the zone cluster. Set the resource group affinities as shown in Figure 32. Create the RAC server and RAC listener resources in this scalable resource group with the resource dependencies shown in Figure 32. Bring resource group `DBS` online.

Oracle RAC 9i/10g/11g on hardware RAID

Zone clusters support the ability to run Oracle RAC on hardware RAID, although this configuration is not typical. This configuration uses neither a file system nor ASM.

The configuration for Oracle RAC 10g/11g on hardware RAID is the same as that of Figure 26 ("Oracle RAC 10g/11g on ASM"). The configuration for Oracle RAC 9i would eliminate the RAC Instance Proxy and add in the RAC Server, RAC Listener, and Logical Host resources, as shown in the earlier sections on Oracle RAC 9i configurations.

Multiple Configurations on a Single System

One of the major benefits of zone clusters is the ability to support multiple configurations on a single physical cluster. Any combination of the previously shown configurations can be placed on a single physical cluster. With sufficient hardware resources it is possible to configure all supported configurations on a single cluster. Naturally, the administrator can configure multiple zone clusters using the same type of configuration on a single physical cluster.

In order to demonstrate some important points about multiple zone cluster configurations, Figure 33 depicts an arbitrary example of two separate zone clusters on one physical cluster. Zone Cluster A supports Oracle RAC 10g on Oracle Solaris Volume Manager for Oracle Solaris Cluster and Zone Cluster B supports Oracle RAC 9i on Oracle Solaris Volume Manager for Oracle Solaris Cluster. As this example shows, the different zone clusters can support different versions of Oracle RAC.

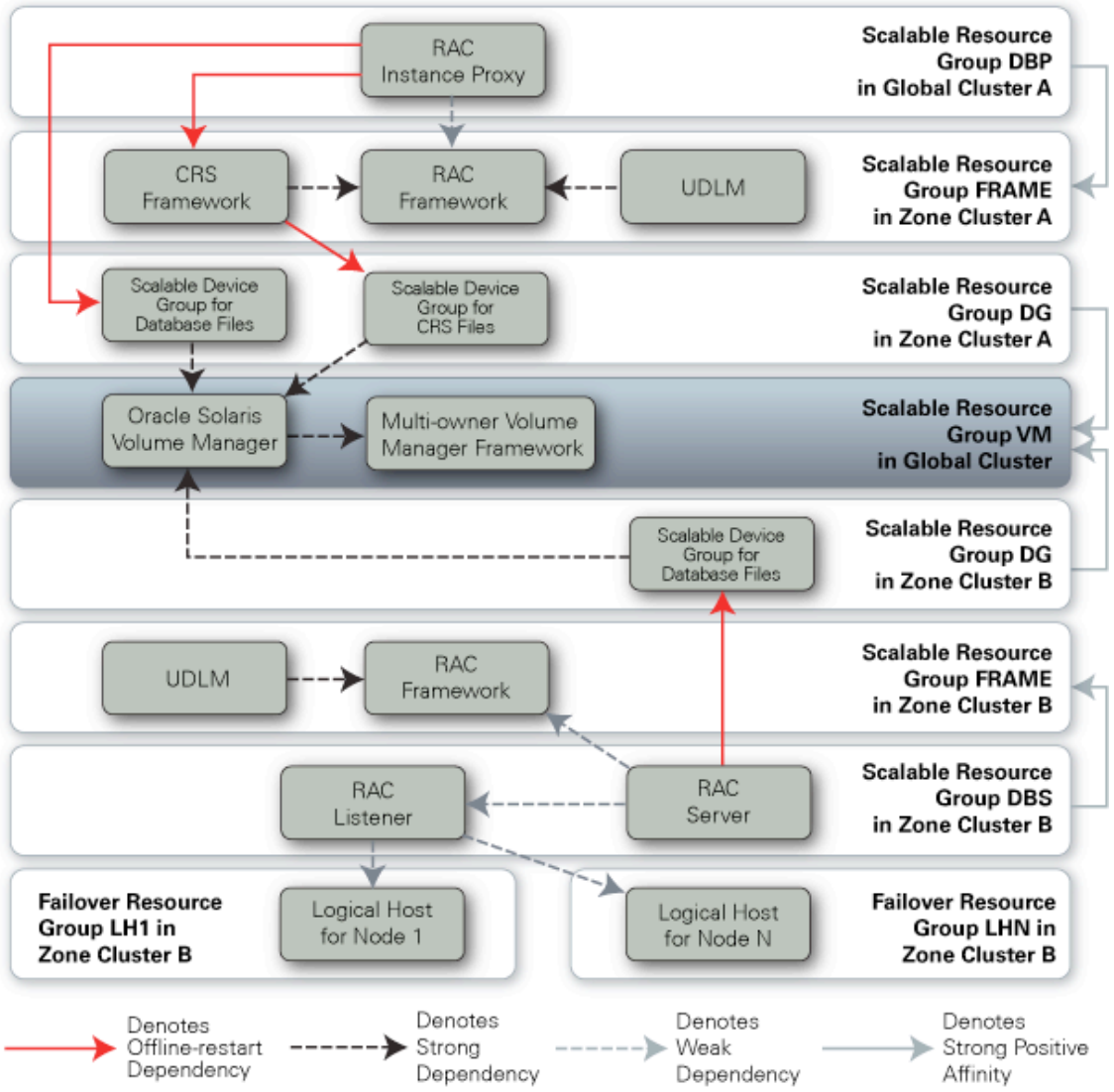

Figure 33. Resource relationships for Oracle RAC 10g on Oracle Solaris Volume Manager and Oracle RAC 9i on Oracle Solaris Volume Manager in different zone clusters.

Applications and resources within a zone cluster are independent of applications and resources in all other zone clusters, unless the global zone administrator specifically sets up a relationship in the form of a dependency or affinity. Names in different zone clusters are independent. Figure 33 shows that both Zone Cluster A and Zone Cluster B have a resource group with the name `FRAME`. There is no name conflict, because these resource groups live in different name spaces. Each zone cluster has its own name space.

Some configurations require support from the global zone. This is true whenever volume managers are used, because volume managers exist only in the global zone at this time. There can only be one RAC framework resource in a zone, and that restriction also applies to the global zone. The RAC framework resource can concurrently support any number of volumes managed by either the Oracle Solaris

Volume Manager or the Veritas Volume Manager cluster feature, which is only supported in the global zone. The RAC framework resource can additionally support concurrently any Oracle RAC software running in that zone. Thus if a RAC framework resource already exists in the global zone for an existing zone cluster, that same RAC framework resource will support the next zone cluster needing such support. Figure 33 shows an example of one RAC framework resource supporting multiple zone clusters.

About the Authors

Gia-Khanh Nguyen is the architect for the Oracle Solaris Cluster Database solutions. He contributed to the product requirement and design specifications for the support of Oracle RAC in the zone cluster project and made demonstrations of the features. He earned an MS in Computer Science and Electrical Engineering from the Engineering School for Informatics, Electronics and Automation, France.

Tim Read is a Software Developer for the Oracle Solaris Cluster Group. His main role is the development of the Oracle Solaris Geographic Edition product. He has written a number of whitepapers and books on high availability and disaster recovery including Oracle Solaris Cluster Essentials, published in 2010. He has a B.Sc. in Physics with Astrophysics from the University of Birmingham in the UK.

The original white paper was co-authored by Dr. Ellard Roush.

Acknowledgments

The development of this feature was done by the project team, and recognition must be extended to all team members who contributed in a variety of ways: Zoram Thanga, Pramod Rao, Tirthankar Das, Sambit Nayak, Himanshu Ashiya, Varun Balegar, Prasanna Kunisetty, Gia-Khanh Nguyen, Robert Bart, Suraj Verma, Harish Mallya, Ritu Agrawal, Madhan Balasubramanian, Ganesh Ram Nagarajan, Bharathi Subramanian, Thorsten Frueauf, Charles Debardeleben, Venkateswarlu Tella, Hai-Yi Cheng, Lina Muryanto, Jagrithi Buddharaja, Nils Pedersen, and Burt Clouse.

References

TABLE 4. REFERENCES

WEB SITES	
Oracle Solaris Cluster	http://www.oracle.com/us/products/servers-storage/solaris/cluster-067314.html http://www.oracle.com/technetwork/server-storage/solaris-cluster/index.html
Oracle Partitioning Rules and Oracle Solaris 10 Containers	http://www.oracle.com/us/corporate/pricing/partitioning-070609.pdf
Oracle Database Support for Server Virtualization	http://oracle.com/technology/products/database/clustering/certify/db_virtualization_support.pdf http://www.oracle.com/technetwork/database/clustering/tech-generic-unix-new-166583.html
ORACLE TECHNICAL ARTICLES AND WHITE PAPERS	
Zone Clusters—How to Deploy Virtual Clusters and Why	http://www.oracle.com/technetwork/server-storage/solaris-cluster/documentation/zone-clusters-why-whitepaper-322070.pdf
Oracle Solaris Cluster Essentials	Prentice Hall ISBN-13 978-0-13-248622-4
DOCUMENTATION	
Oracle Solaris Cluster 3.3 Documentation Center	http://download.oracle.com/docs/cd/E19680-01/html/821-1261/fxiwa.html
Oracle Solaris Cluster Concepts Guide for Oracle Solaris OS	http://download.oracle.com/docs/cd/E19680-01/html/821-1254/index.html
Oracle Solaris Cluster System Administration Guide for Oracle Solaris OS	http://download.oracle.com/docs/cd/E19680-01/html/821-1257/index.html
System Administration Guide: Oracle Solaris Containers—Resource Management and Oracle Solaris Zones	http://download.oracle.com/docs/cd/E18752_01/html/817-1592/index.html
Oracle Solaris Cluster Software Installation Guide for Oracle Solaris OS, “Configuring a Zone Cluster”	http://download.oracle.com/docs/cd/E19680-01/html/821-1255/gqzen.html
Oracle Solaris Cluster Data Service for Oracle RAC Guide for Oracle Solaris OS	http://download.oracle.com/docs/cd/E19680-01/html/821-1262
`clzonecluster(1CL)` man page	http://download.oracle.com/docs/cd/E19680-01/html/821-1263/clzonecluster-1cl.html
`sysidcfg(4)` man page	http://download.oracle.com/docs/cd/E19253-01/816-

	5174/sysidcfg-4/index.html
`zoneadm(1M)` man page	http://download.oracle.com/docs/cd/E19253-01/816-5166/zoneadm-1m/index.html
`zonecfg(1M)` man page	http://download.oracle.com/docs/cd/E19253-01/816-5166/zonecfg-1m/index.html

Oracle is committed to developing practices and products that help protect the environment

Deploying Oracle Real Application Clusters on Oracle Solaris Zones
July 2011
Authors: Gia-Khanh Nguyen, Tim Read

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

SOFTWARE. HARDWARE. COMPLETE.