Business Continuity with the Cisco Unified Computing System and EMC Symmetrix VMAX

Table of Contents

Executive Summary

EMC and Cisco recently completed a joint project to demonstrate the high availability (HA), scaling and disaster recovery (DR) capabilities of a combined compute and storage architecture. The initial objective was to validate current established solutions for local HA and DR with the Cisco Unified Computing System™ (UCS) and its new architectural paradigm. EMC and Cisco demonstrated that the current HA and DR design and solution paradigms do not need to be changed with the new compute-tier architecture that the Cisco Unified Computing System provides, and furthermore, using certain EMC Symmetrix VMAX and Cisco Unified Computing System technologies can actually increase the value of current designs for high availability and disaster recovery.

The test was conducted in two phases. The first phase of the project studied the HA resiliency of the combined architecture as well as the ease of adding compute nodes to a running Oracle Real Application Clusters (RAC) instance and how the Cisco Unified Computing System construct of service profiles (stateless computing) makes this task much easier and faster.

The second phase of the project focused on the steps necessary to achieve business continuity in the event of a disaster. This part of the endeavor used two sets of Cisco Unified Computing System domains and VMAX systems separated by distances that can be supported by synchronous replication. The disaster was induced by injecting failure that simulates the loss of a building. This document describes the process to bring applications online at the DR site after a planned or unplanned event.

This document assumes that the reader has an architectural understanding of the Cisco Unified Computing System and the service profile concept, EMC's VMAX storage system, and related software.

Business Problems Addressed

Designing High Availability and Disaster Recovery into Private Clouds

Organizations are moving away from direct-attach models of compute, network, and storage resources for applications toward a consolidated and shared infrastructure. There are many variations on this new approach. We will not discuss all the variants in detail in this paper, but rather show how adopting technologies built for these new models can be used to provide critical functions such as high availability, ease of scaling (adding resources), and disaster recovery.

One aspect of the new architectural paradigm that is not frequently discussed is that the adoption of any new model or architecture necessitates a reassessment of standard DR practices. This document therefore attempts to answer this question: "What will happen to my current HA and DR solutions and designs if I move to a private cloud compute model?"

Increased Utilization of Disaster Recovery Assets

Better use of DR assets is a critical goal in data centers. Frequently, DR sites and the equipment associated with them remain idle, awaiting a future disaster that may never happen. To complicate matters, many upper-level managers have instructed their IT teams to have two or more "hot," or active, sites. Fortunately, new technology such as the Cisco Unified Computing System provides the capability to repurpose DR equipment in an event-driven model in a matter of hours. When a DR event occurs, the hardware assets can simply be re-provisioned with new service profiles from the primary site, and application services can be restored. Dedicated, spare equipment is no longer required.

Solution Overview

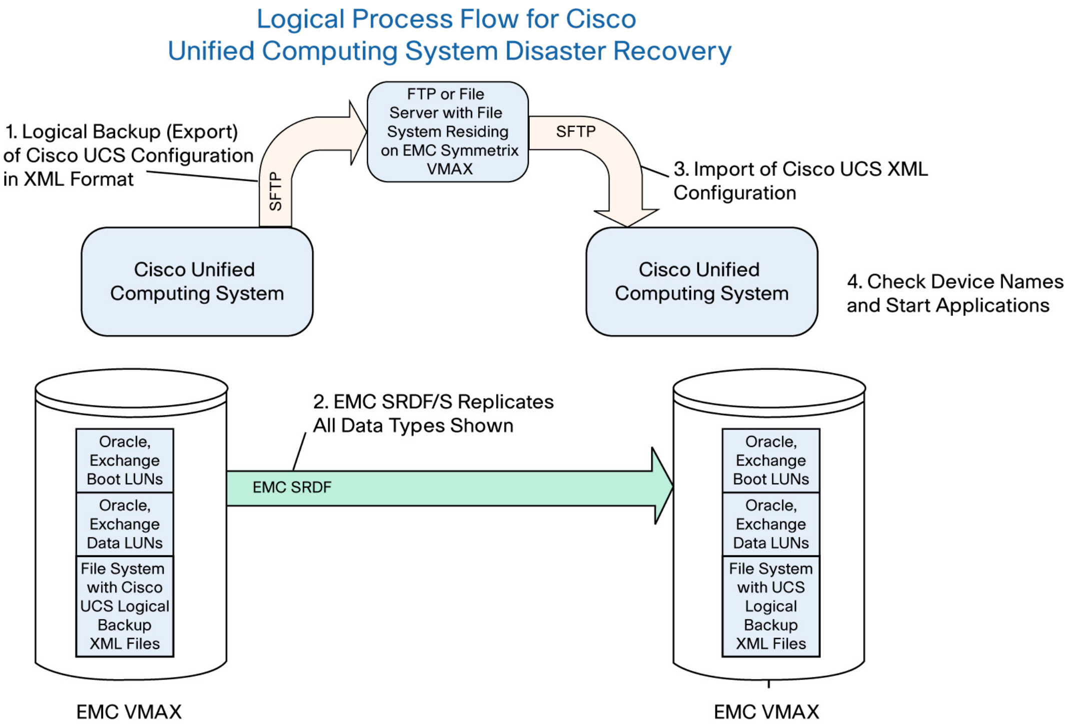
Figure 1 shows the overall DR solution built by following Cisco and EMC VMAX best practices. Subsequent sections of the document will describe both the HA and DR phases of the test in detail.

The DR solution has two main components: logical backups of the Cisco Unified Computing System XML configuration, and EMC Symmetrix Remote Data Facility (SRDF), EMC's synchronous replication technology. The entire configuration of the Cisco Unified Computing System is contained in a hierarchical XML schema. The configuration can be accessed through an API from either the Cisco® UCS Manager command-line interface (CLI) or the Cisco UCS Manager GUI.

Cisco UCS Manager provides the capability to back up the current running configuration, dumping an XML file to a file system outside the fabric interconnects. The transport protocol for moving this XML file off the fabric interconnects to a file system is either FTP, Secure Copy Protocol (SCP), or Secure FTP (SFTP). After the XML file is written to the file system, EMC SRDF replicates the file system containing the XML file automatically to a target file system at the DR location. The XML file can be accessed at the DR site in case of a disastrous event, and the configuration of the Cisco Unified Computing System at the production site can be easily and quickly restored. Furthermore, to restore all applications and services at the disaster recovery site, all other components of the solution - in particular, the operating system and data logical unit numbers (LUNs) - are replicated synchronously using EMC SRDF to the DR site.

With the Cisco Unified Computing System configuration, OS, Oracle binary images, and application data all now available at the DR site, an existing Cisco Unified Computing System configuration can be imported using Cisco UCS Manager, causing the service profiles to be associated to blades at the DR site and to take on those server definitions or personalities.

Figure 1. Process Flow for Overall Disaster Recovery Solution


Logical Process Flow for Cisco Unified Computing System Disaster Recovery
1. Logical Backup (Export) of Cisco UCS Configuration in XML Format
SFTP
FTP or File Server with File System Residing on EMC Symmetrix VMAX
SFTP
3. Import of Cisco UCS XML Configuration
Cisco Unified Computing System
Cisco Unified Computing System
4. Check Device Names and Start Applications
2. EMC SRDF/S Replicates All Data Types Shown
EMC SRDF
Oracle, Exchange Boot LUNs
Oracle, Exchange Data LUNs
File System with Cisco UCS Logical Backup XML Files
Oracle, Exchange Boot LUNs
Oracle, Exchange Data LUNs
File System with UCS Logical Backup XML Files
EMC VMAX
EMC VMAX

Main Technologies Used in the Solution

The joint solution uses several critical technologies that warrant specific attention. Figure 1 serves as a reference to show where these technologies fit into the overall solution.

Cisco Unified Computing System
Service Profiles and Disaster Recovery

A service profile is a logical abstraction of a physical server’s identity. Service profiles can be moved from blade to blade as necessary, each time reprogramming the hardware so that it takes on the attributes defined in the profile. For a more in-depth explanation of service profiles, see Understanding Cisco Unified Computing System Service Profiles. Technical documentation found here.

Because service profiles are in XML format, they can be moved between Cisco Unified Computing Systems using the Cisco UCS Manager export and import functions. The profiles allow a blade at the DR site to be fully reprogrammed and reprovisioned so that it functions as the exact same server that was running at the primary site.

Current or legacy blade server architectures require the IT staff to maintain the exact same configuration for the primary and secondary servers. Any configuration “drift” can cause the disaster recovery assets to become unusable by the application upon failover, thus severely affecting business continuity.

Exporting and Importing the Configuration (XML)

The Cisco Unified Computing System is an integrated array of compute nodes behind a pair of intelligent server controllers. This new layer of abstraction must itself be protected in the event of a disaster. The Cisco Unified Computing System provides a means to do this by backing up the system configuration and importing it on the original or a new Cisco Unified Computing System. Each Cisco Unified Computing System or pair of fabric interconnects is typically referred to as a domain and can span multiple chassis. The Cisco Unified Computing System uses an export-import model to achieve cross-domain service profile mobility. This mobility allows the server to be seen by the operating system and the application to be reinstituted at the DR site.

Setting Boot-from-SAN Policies

A service profile can contain many different policies that are associated to a given profile. One of these policies is the boot policy, which essentially programs the system’s converged network adapter (CNA) firmware and instructs it where to look for the master boot record on the boot device. The policy is extremely flexible and can provide up to four different targets to search for the boot LUN. The solution presented in this document consists of a boot policy that has the secondary or alternate boot LUNs as the LUNs that exist on the VMAX arrays at the disaster recovery site. This policy allows the same boot policy to be used at both the primary and secondary sites (after import) without any changes. The service profile, in combination with the boot policies’ extensive flexibility, masks the complexities of DR Redundant SAN zoning is performed on both the local and remote sides to make this process possible.

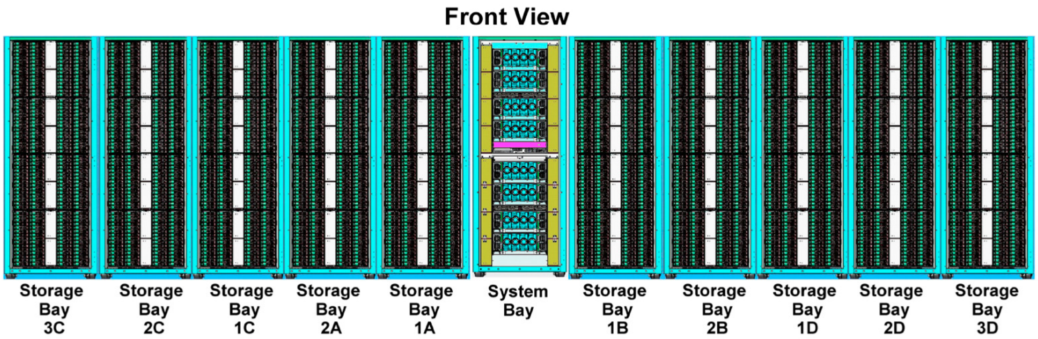
Note that the Cisco Unified Computing System does not formally require OS images to be SAN-resident. However, they must be SAN-resident to achieve the stateless aspect of the Cisco Unified Computing System in general, and SAN-resident OS images are mandatory for the solution described in this document.

EMC Symmetrix VMAX

The EMC Symmetrix VMAX system with the Enginuity operating environment is a new enterprise-class storage array that is built on the strategy of simple, intelligent, modular storage. The array incorporates a new scalable fabric interconnect design that allows the storage array to provide improved performance and scalability for demanding enterprise storage environments while maintaining support for EMC’s broad portfolio of platform software offerings. The storage array can seamlessly grow from an entry-level configuration with a single, highly available VMAX Engine and 1 storage bay into the world’s largest storage system with 8 engines and 10 storage bays. The largest supported configuration is shown in Figure 2. The figure also shows the range of configurations supported by the EMC Symmetrix VMAX storage array.

Figure 2. EMC Symmetrix VMAX System Features

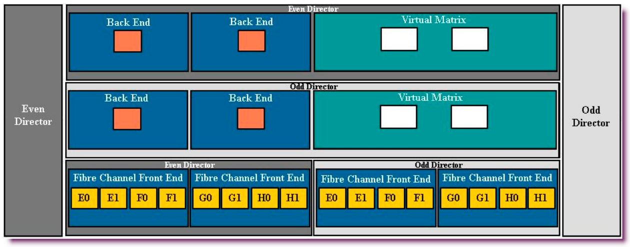
- 2 to 16 Director Boards
- 40 to 2400 Disk Drives
- Up to 2 Petabytes (PB) Usable Capacity
- Up to 128 Fibre Channel Ports
- Up to 64 FICON Ports
- Up to 64 Gigabit Ethernet and iSCSI Ports
- Up to 1 Terabyte (TB) Global Memory



The enterprise-class applications in a modern data center are expected to be always available. The design of the EMC Symmetrix VMAX storage array enables it to meet this stringent requirement. The hardware and software architecture of the EMC Symmetrix VMAX storage array allows capacity and performance upgrades to be performed online with no negative effect on production applications. In fact, all configuration changes, hardware and software updates, and service procedures are designed to be performed online and nondisruptively. This capability helps ensure that customers can consolidate without compromising availability, performance, or functions, while taking advantage of true pay-as-you-grow economics for high-growth storage environments.

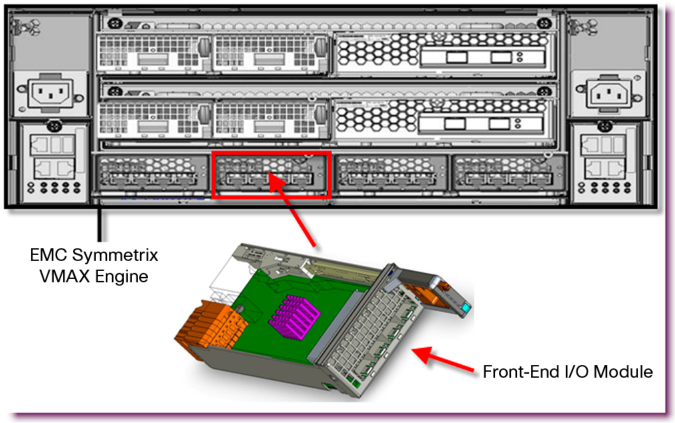
EMC Symmetrix VMAX can include 2 to 16 directors inside 1 to 8 EMC Symmetrix VMAX Engines. Each VMAX Engine has its own redundant power supplies, cooling fans, Standby Power Supply (SPS) modules, and environmental modules. Furthermore, the connectivity between the Symmetrix VMAX array engines provides a direct connection from each director to every other director, creating a virtual matrix. Each VMAX Engine has two directors that can offer up to 8 ports each, therefore allowing up to 16 ports per VMAX Engine. Figure 3 shows the typical setup of a Fibre Channel-only EMC Symmetrix VMAX engine.

Figure 3. EMC Symmetrix VMAX Engine



In Symmetrix storage arrays, the directors were set up differently: front-end directors, back-end directors, and memory boards were separate entities. In Symmetrix VMAX, these have been combined into integrated directors to increase the number of possible directors in the Symmetrix array. Each integrated director has three parts: the back-end director, the front-end director, and the cache memory module. The back-end director consists of two back-end I/O modules with four logical directors A, B, C, and D (not labeled in the figure) that connect directly into the integrated director. The front-end director consists of two front-end I/O modules with four logical directors (labeled E, F, G, and H) that are located in the corresponding I/O annex slots. The front-end I/O modules are connected to the director through the midplane (Figure 4).

Figure 4. Front-End I/O Placement



The cache memory modules are located within each integrated director, each with eight available memory slots. Memory cards range from 2 to 8 GB, consequently allowing from 16 to 64 GB per integrated director.

Note that mixing memory card sizes within a director is not allowed; they must all be 2 GB, 4 GB, or 8 GB in size.

For added redundancy, Symmetrix VMAX employs the use of mirrored cache, so memory is mirrored across engines in a multiple-engine setup. In a single-engine configuration, the memory is mirrored inside the engine over the two integrated directors.

Table 1 shows the drive types that Symmetrix VMAX supports.

Table 1. Disk-Drive Support

Drive Type	Rotational Speed	Capacity
4-Gbps Fibre Channel	15,000 rpm	146, 300, and 450 GB
4-Gbps Fibre Channel	10,000 rpm	400 GB
SATA	7,200 rpm	1 TB
4-Gbps flash memory (solid-state drive [SSD])	-	200 and 400 GB

A more detailed discussion of the Symmetrix VMAX storage array can be found in the product guide available on Powerlink®, EMC's customer- and partner-only extranet.

EMC PowerPath

EMC PowerPath is host-based software that provides path management.

PowerPath operates with several storage systems, on several operating systems, with Fibre Channel and iSCSI data channels (and, with Microsoft Windows 2000 and Windows Server 2003 only, parallel SCSI channels).

EMC PowerPath works with the storage system to intelligently manage I/O paths. It supports multiple paths to a logical device, enabling PowerPath to provide:

- Automatic failover in the event of a hardware failure. PowerPath automatically detects path failure and redirects I/O to another path.

- Dynamic multipath load balancing. PowerPath distributes I/O requests to a logical device across all available paths, thus improving I/O performance and reducing management time and downtime by eliminating the need to configure paths statically across logical devices.

EMC PowerPath Features

- Multiple paths, for higher availability and performance:
  - PowerPath supports multiple paths between a logical device and a host. Having multiple paths enables the host to access a logical device even if a specific path is unavailable. Also, multiple paths can share the I/O workload to a given logical device.
  - Dynamic multipath load balancing: PowerPath improves a host's ability to manage heavy I/O loads by continually balancing the load on all paths, eliminating the need for repeated static reconfigurations as workloads change.
  - Proactive I/O path testing and automatic path recovery: PowerPath periodically tests failed paths to determine whether they have been fixed. If a path passes the test, the path is restored automatically, and
  - PowerPath resumes sending I/O to it. During path restoration, the storage system, host, and application remain available.
  - PowerPath also periodically tests live paths that are idle. This allows PowerPath to report path problems quickly, avoiding delays that would otherwise result from trying to use a defective path when I/O is started on the logical device.
- **Automatic path failover** - PowerPath automatically redirects I/O from a failed path to an alternate path. This eliminates loss of data and application downtime. Failovers are transparent and nondisruptive to applications.
- **High-availability cluster support** - PowerPath is particularly beneficial in cluster environments, as it can prevent operational interruptions and costly downtime. PowerPath's path failover capability avoids node failover, maintaining uninterrupted application support on the active node in the event of a path disconnect (as long as another path is available).

EMC Symmetrix Remote Data Facility (SRDF)

Symmetrix Remote Data Facility/Synchronous (SRDF/S) is a remote replication product that maintains a real-time (synchronous) mirrored copy of data in physically separated Symmetrix systems within an SRDF configuration.

SRDF/S offers the following major features and benefits:

- High data availability
- High performance
- Flexible configurations
- Host and application software transparency
- Automatic recovery from a failure
- Significantly reduced recovery time after a disaster
- Reduced backup and recovery costs
- Reduced disaster recovery complexity and less planning and testing

The SRDF/S operation is transparent to the host operating system and host applications. It does not require additional host software for duplicating data on the participating Symmetrix units. SRDF offers greater flexibility through additional modes of operation.

EMC TimeFinder

Symmetrix TimeFinder is essentially a business continuance solution that allows you to use special business continuance volume (BCV) Symmetrix devices. Copies of data from a standard Symmetrix device (which are online for regular I/O operations from the host) are sent and stored on BCV devices to mirror the primary data. Uses for the BCV copies can include backup, restore, decision support, and application testing. Each BCV device has its own host address and is configured as a stand-alone Symmetrix device.

The following products make up the EMC TimeFinder family:

- TimeFinder/Mirror: For general monitor and control operations
- TimeFinder/Snap: For snap copy sessions
- TimeFinder/Clone: For clone copy sessions
- TimeFinder/CG: For consistency groups

Testing Approach

Two well-known commercial applications, Oracle RAC 11G R1 and Microsoft Exchange 2007, were used to drive the load on the hardware and test the failure use cases.

The EMC lab in Santa Clara, California, houses two VMAX storage arrays and two Cisco Unified Computing System domains, which were used for the testing. The Cisco Unified Computing System domains were running the Version 1.0(2d) firmware. Oracle RAC was used to demonstrate the Cisco Unified Computing System and EMC advantages for availability and scaling out the cluster. The RAC cluster used Oracle Enterprise Linux (OEL) 5.3 as the operating system and Oracle's own clustering software, Cluster Ready Services (CRS). Traditional (active/passive) failover was also demonstrated using a Microsoft Exchange cluster with Microsoft Cluster Server (MSCS) as the clustering framework. However, the test primarily focused on the behavior of the Oracle RAC environment in different failure scenarios, discussed extensively in the rest of this document.

The remainder of this document is divided into two parts: Part 1 discusses HA and scaling (adding a node) testing; Part 2 focuses on disaster recovery. In both parts, the failures were generated in the different topologies and results recorded for both Oracle RAC and Microsoft Exchange. The motivation for the project was to obtain an application-centric view of the test cases.

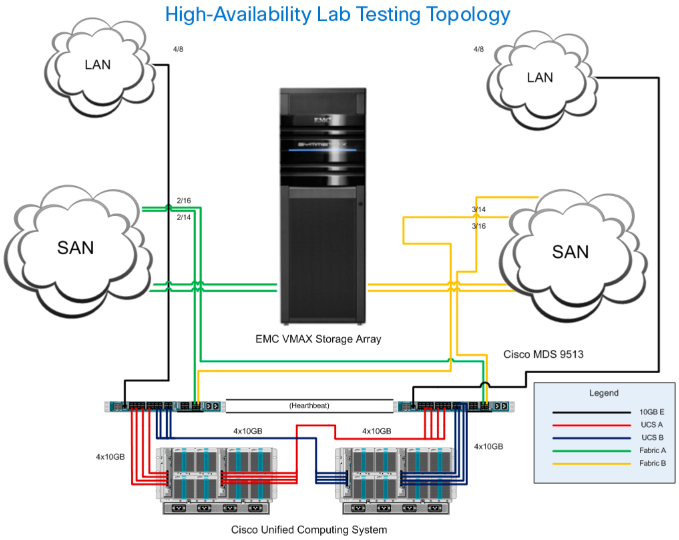
Swingbench was used to generate load against the Oracle RAC database. It was run against both RAC nodes at the same time to show continuity of service for each instance.

Part 1: Local High Availability (HA) and Scaling

Description, Topology, and Assumptions

The first phase of the testing involved a single Cisco Unified Computing System domain (a pair of Cisco UCS 6120XP 20-Port Fabric Interconnects) and two chassis. Figure 5 shows the architecture. Oracle RAC was installed on two blades. The two RAC nodes were in separate chassis to demonstrate accepted HA design principles. RAC scaling through the addition of a new node was shown by associating the existing RAC service profile to an additional blade. Microsoft Exchange testing consisted of a single Exchange node in each chassis along with a blade serving as the Exchange 2007 Hub Transport server.

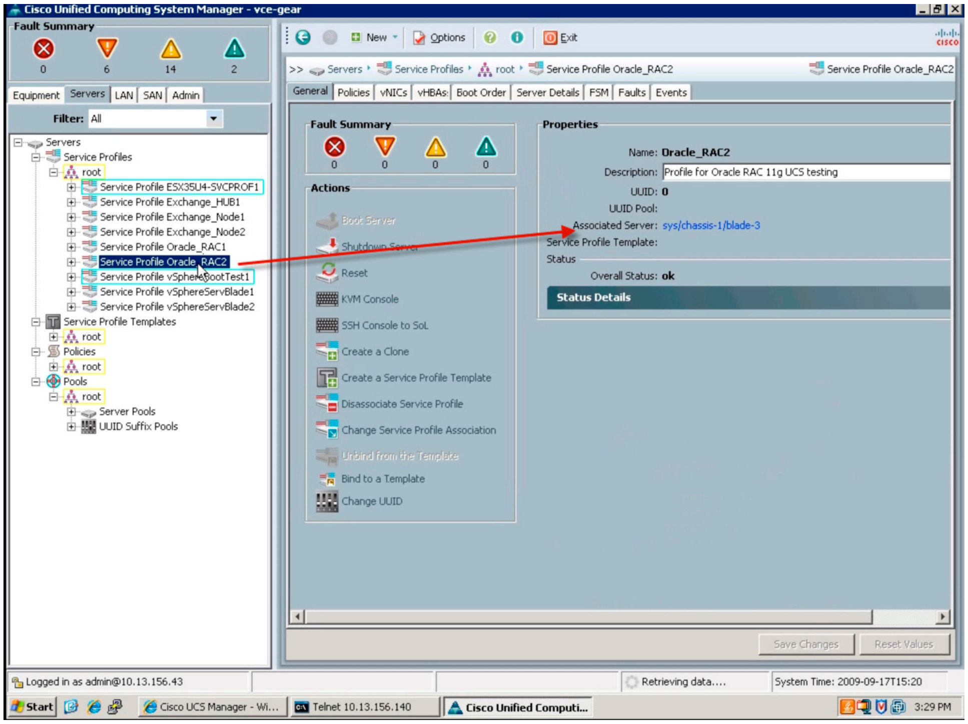
Figure 5. Local HA and Scaling Topology



System Behavior During Normal Operation

The baseline configuration for Oracle RAC was a two-node cluster running on blades in two different chassis to remove the chassis itself from being a single point of failure (SPOF). Figure 6 shows the configuration of the servers at the production site as viewed from the Cisco UCS Manager GUI.

Figure 6. Association of Service Profiles to Blades

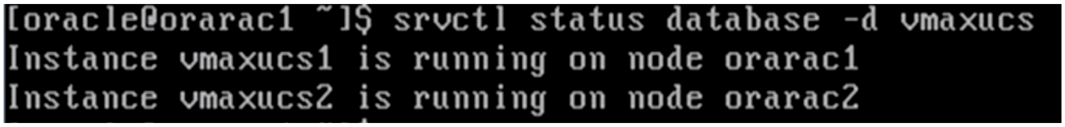

Cisco Unified Computing System Manager - vce-gear
Fault Summary
0
6
14
2
Equipment
Servers
LAN
SAN
Admin
Filter: All
Servers
Service Profiles
root
Service Profile ESX35U4-SVCPROF1
Service Profile Exchange_HUB1
Service Profile Exchange_Node1
Service Profile Exchange_Node2
Service Profile Oracle_RAC1
Service Profile Oracle_RAC2
Service Profile vSphereBootTest1
Service Profile vSphereServBlade1
Service Profile vSphereServBlade2
Service Profile Templates
root
Policies
root
Pools
root
Server Pools
UUID Suffix Pools
New
Options
Exit
>> Servers › Service Profiles › root › Service Profile Oracle_RAC2
Service Profile Oracle_RAC2
General
Policies
vNICs
vHBAs
Boot Order
Server Details
FSM
Faults
Events
Fault Summary
0
0
0
0
Actions
Boot Server
Shutdown Server
Reset
KVM Console
SSH Console to SoL
Create a Clone
Create a Service Profile Template
Disassociate Service Profile
Change Service Profile Association
Unbind from the Template
Bind to a Template
Change UUID
Properties
Status Details
Save Changes
Reset Values
Start
Cisco UCS Manager - Wi...
Telnet 10.13.156.140
Cisco Unified Computi...
3:29 PM

Name: Oracle_RAC2
Description: Profile for Oracle RAC 11g UCS testing
UUID: 0
UUID Pool:
Associated Server: sys/chassis-1/blade-3
Service Profile Template:
Status
Overall Status: ok

Logged in as admin@10.13.156.43
Retrieving data....
System Time: 2009-09-17T15:20

As shown in Figure 7, from the perspective of Oracle RAC, the database named vmaxucs is running on two nodes: orarac1 and orarac2. These hosts are physical manifestations of the service profiles, Oracle_RAC1 and Oracle_RAC2, shown in Figure 6.

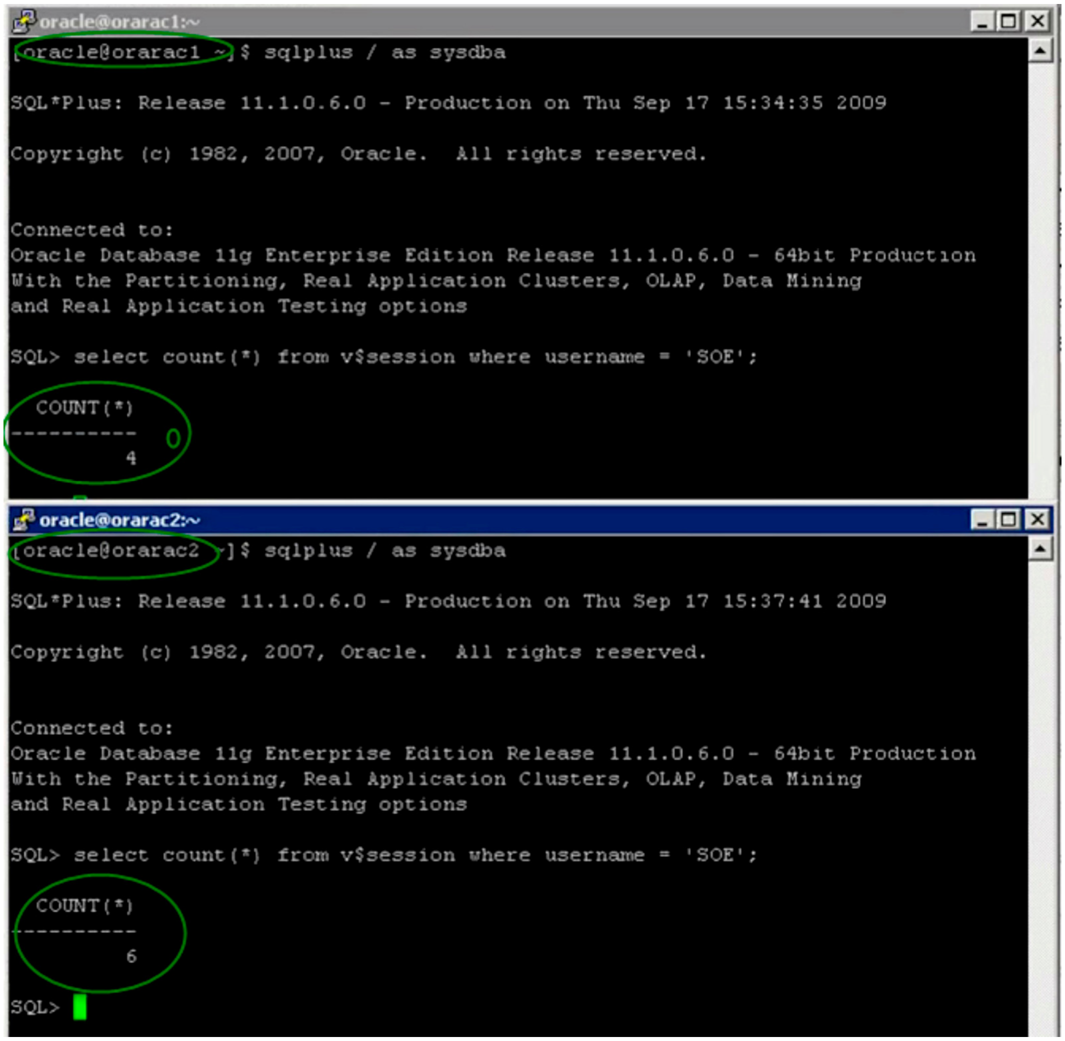
Figure 7. Status of Oracle RAC Instances



```
[oracle@orarac1 ~]$ srvctl status database -d vmaxucs
Instance vmaxucs1 is running on node orarac1
Instance vmaxucs2 is running on node orarac2
```

In addition to the two-node Oracle RAC configuration, as shown in Figure 8, 10 Swingbench users are connected to the database and distributed across the two nodes.

Figure 8. Distribution of Users Across Oracle RAC Nodes

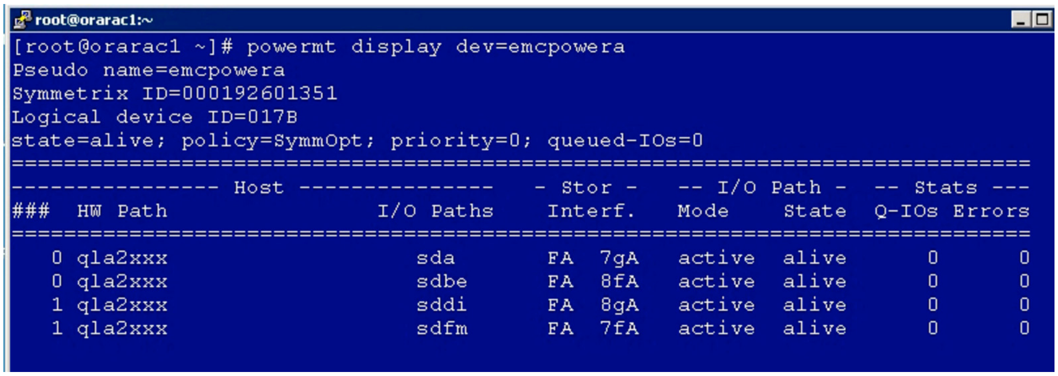


```
[oracle@orarac1 ~]$ sqlplus / as sysdba

SQL*Plus: Release 11.1.0.6.0 - Production on Thu Sep 17 15:34:35 2009

Copyright (c) 1982, 2007, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 - 64bit Production
With the Partitioning, Real Application Clusters, OLAP, Data Mining
and Real Application Testing options

SQL> select count(*) from v$session where username = 'SOE';

  COUNT(*)
----------
         4
```

```
[oracle@orarac2 ~]$ sqlplus / as sysdba

SQL*Plus: Release 11.1.0.6.0 - Production on Thu Sep 17 15:37:41 2009

Copyright (c) 1982, 2007, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 - 64bit Production
With the Partitioning, Real Application Clusters, OLAP, Data Mining
and Real Application Testing options

SQL> select count(*) from v$session where username = 'SOE';

  COUNT(*)
----------
         6

SQL>
```

The Oracle RAC nodes access the application devices using four Fibre Channel paths: two from each of the Cisco UCS 6100 fabric interconnects. Figure 9 shows this configuration from a PowerPath perspective for one of the devices.

Figure 9. EMC PowerPath Command Display Output



```
[root@oraracl ~]# powermt display dev=emcpowera
Pseudo name=emcpowera
Symmetrix ID=000192601351
Logical device ID=017B
state=alive; policy=SymmOpt; priority=0; queued-IOs=0
==============================================================================
--------------- Host ---------------   - Stor -   -- I/O Path -   -- Stats ---
###  HW Path                I/O Paths    Interf.   Mode     State   Q-IOs Errors
==============================================================================
   0 qla2xxx                   sda       FA  7gA   active   alive       0      0
   0 qla2xxx                   sdbe      FA  8fA   active   alive       0      0
   1 qla2xxx                   sddi      FA  8gA   active   alive       0      0
   1 qla2xxx                   sdfm      FA  7fA   active   alive       0      0
```

Failure of Fibre Channel Uplink

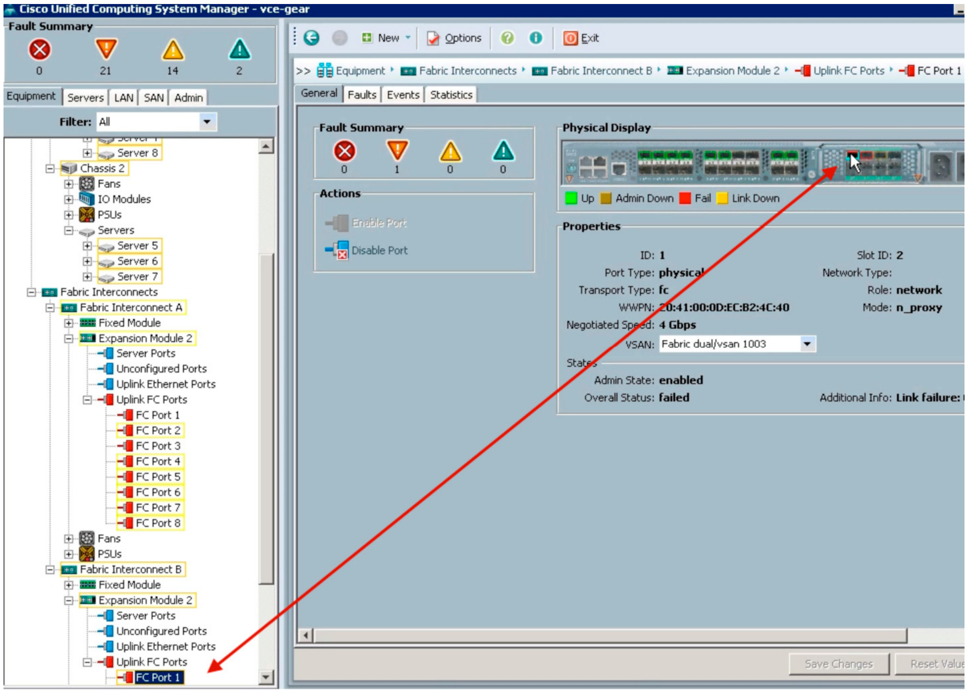
The goal of this test was to study the effects of an upstream Fibre Channel SAN path failure on the applications. The failure was simulated by abruptly turning off the Fibre Channel port on the Cisco Fibre Channel switch to which the uplink ports on the fabric interconnect were connected.

Results

EMC PowerPath software handled the loss of the paths and rerouted all outstanding I/O operations down the remaining path to both the operating system and the application data LUNs.

The link failures were induced by using Cisco UCS Manager and manually disabling the Fibre Channel uplink ports in the expansion module. Figures 10 and 11 show the status of the uplink port after the simulation of the failure event. The figures show that the operational state of the port is "enabled"; however, the overall status is "failed." This is the case since the failure in the Fibre Channel fabric occurred upstream of the fabric interconnect switches.

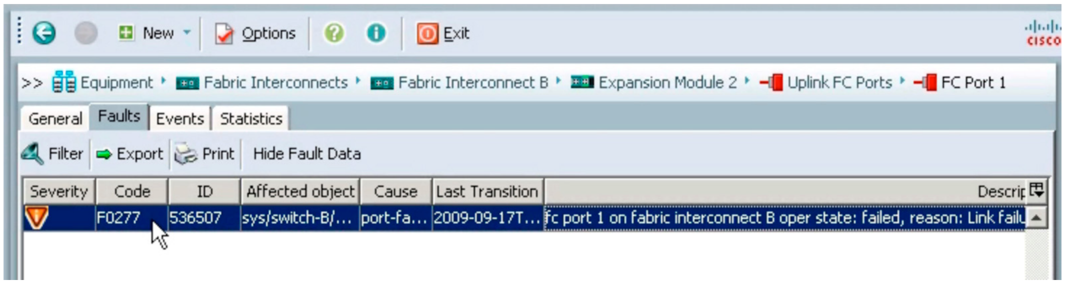
Figure 10. Cisco Unified Computing System SAN Tab and Fibre Channel Port Attributes



ID: **1**
Port Type: **physical**
Transport Type: **fc**
WWPN: **20:41:00:0D:EC:B2:4C:40**
Negotiated Speed: **4 Gbps**
VSAN: Fabric dual/vsan 1003
Slot ID: **2**
Network Type:
Role: **network**
Mode: **n_proxy**

States
Admin State: **enabled**
Overall Status: **failed**
Additional Info: **Link failure:**

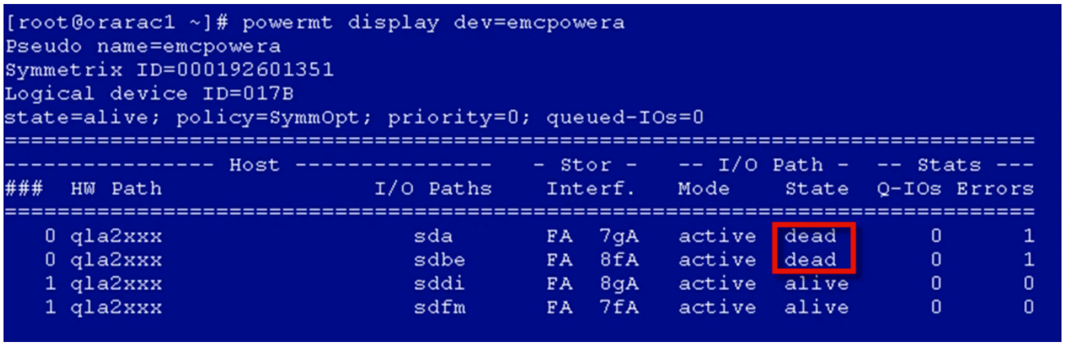
Figure 11. EMC PowerPath GUI Failure Notification


New
Options
Exit
>> Equipment ▸ Fabric Interconnects ▸ Fabric Interconnect B ▸ Expansion Module 2 ▸ Uplink FC Ports ▸ FC Port 1
General
Faults
Events
Statistics
Filter
Export
Print
Hide Fault Data

Severity	Code	ID	Affected object	Cause	Last Transition	Descrip
▽	F0277	536507	sys/switch-B/…	port-fa…	2009-09-17T…	fc port 1 on fabric interconnect B oper state: failed, reason: Link failu

Figure 12 shows the state of the paths for device named emcpowera. The figure shows that the paths have indeed failed from the perspective of EMC PowerPath.

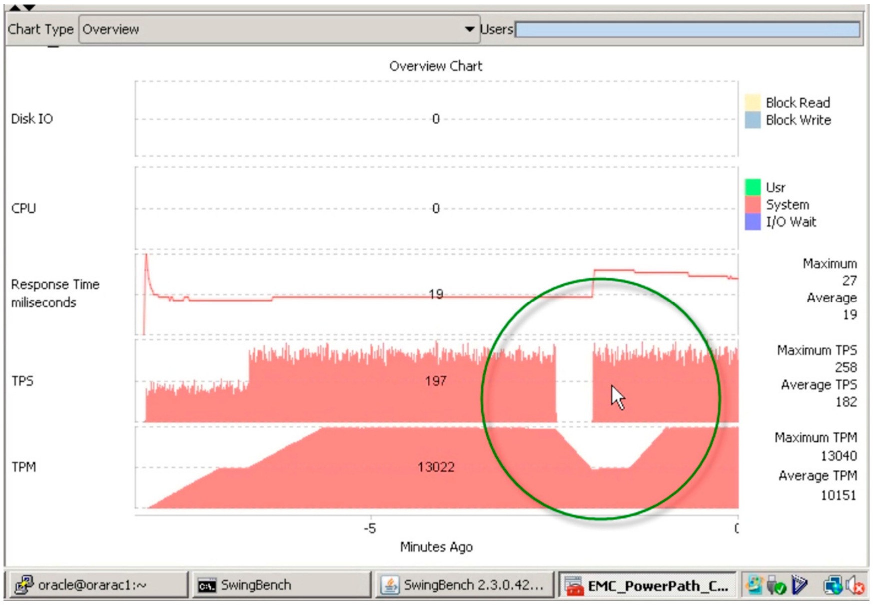
Figure 12. EMC PowerPath CLI Output Showing Failed Paths



```
[root@ronarac1 ~]# powermt display dev=emcpowera
Pseudo name=emcpowera
Symmetrix ID=000192601351
Logical device ID=017B
state=alive; policy=SymmOpt; priority=0; queued-IOs=0
==============================================================================
---------------- Host ---------------   - Stor -   -- I/O Path -   -- Stats ---
###  HW Path                I/O Paths    Interf.   Mode    State   Q-IOs Errors
==============================================================================
   0 qla2xxx                   sda       FA  7gA   active  dead        0      1
   0 qla2xxx                   sdbe      FA  8fA   active  dead        0      1
   1 qla2xxx                   sddi      FA  8gA   active  alive       0      0
   1 qla2xxx                   sdfm      FA  7fA   active  alive       0      0
```

Figure 13 shows the Swingbench GUI and the effect of the Fibre Channel uplink failure on the workload. The figure shows that the workload quickly resumes after PowerPath fails over any I/O processing that had been routed down the failed paths. There is a small pause in the workload as shown in the green circle in the figure. This pause occurs because the operating system becomes momentarily unresponsive as PowerPath marks the failed paths to the boot LUN as failed. The behavior shown in Figure 13 is consistent with PowerPath's multipathing capabilities, and it also shows that the Cisco Unified Computing System architecture does not degrade these capabilities in any way.

Figure 13. Swingbench GUI Display of Transaction Load



Failure of Node (Blade)

A failure of a compute node is induced by powering off a blade and removing it from the chassis. This simulates multiple failure scenarios, including the following:

- Failure of Intel Xeon 5500 series CPU
- Failure of DIMM(s)
- Failure of the Cisco UCS B200 M1 Blade Server itself from various hardware board failures
- Failure of the converged network adapter (CNA)
- Failure of a chassis

Expected Outcomes

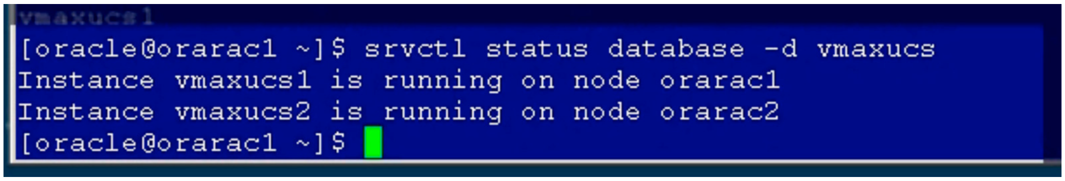
A loss of a compute node in an Oracle RAC instance should result in a loss of connection to all users connected to that server. Depending on the design and configuration of the RAC application, the users will either automatically reconnect to one of the surviving nodes or have to manually reconnect to one of the surviving instances. Nonetheless, users connected to the surviving nodes should not be affected by the failure of a single server.

For active/passive clusters, such as the Microsoft Exchange 2007 cluster that was used in the test, the failure of a compute node should cause one of the surviving nodes to automatically mount the devices from the failed application and restart the failed services. Furthermore, Exchange users who were disconnected due to the failure of the node should automatically be reconnected to the application service after it is successfully restarted on the surviving node.

The test did not use the Oracle Transparent Application Failover (TAF) via the Oracle Call Interface (OCI) C-code-based API; thus, Oracle Data Manipulation Language (DML) statements had to be rolled back and reissued on a surviving instance, resulting in a brief interruption of service. Therefore, this test expected users connected to the surviving node to continue operating normally; for users who were connected to the failed node, execution of the operations they were performing would stop.

This test used the same baseline configuration presented before with an instance running on two blades in two different chassis. For clarity, the normal running state of the Oracle RAC instance is again shown, in Figure 14, by using an Oracle command.

Figure 14. Oracle RAC Instance Distribution on Nodes

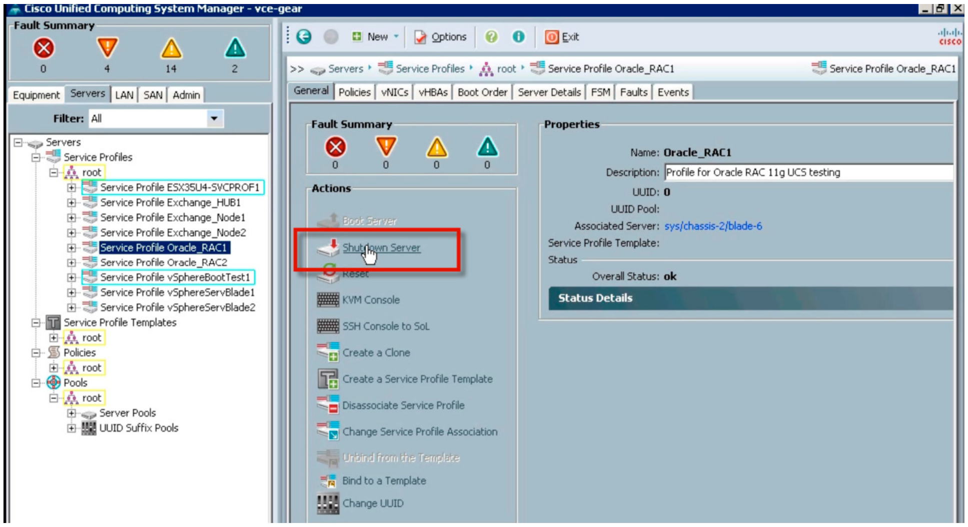


```
vmaxucs1
[oracle@oraracl ~]$ srvctl status database -d vmaxucs
Instance vmaxucs1 is running on node oraracl
Instance vmaxucs2 is running on node orarac2
[oracle@oraracl ~]$
```

Results

One of the blades is made to fail by powering down the blade from the Cisco UCS Manager GUI, as shown in Figure 15.

Figure 15. Cisco UCS Manager Shutting Down a Server

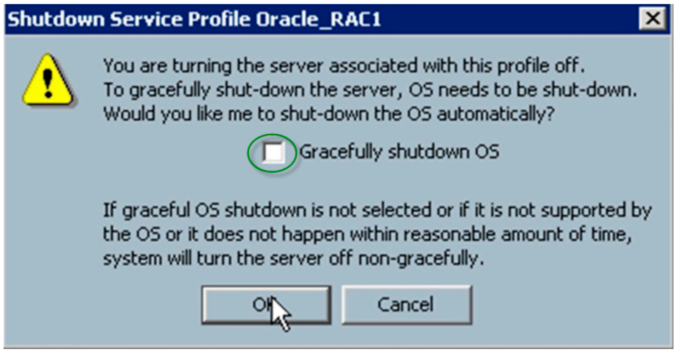


Cisco Unified Computing System Manager - vce-gear
Fault Summary
0
4
14
2
Equipment
Servers
LAN
SAN
Admin
Filter: All
Servers
Service Profiles
root
Service Profile ESX35U4-SVCPROF1
Service Profile Exchange_HUB1
Service Profile Exchange_Node1
Service Profile Exchange_Node2
Service Profile Oracle_RAC1
Service Profile Oracle_RAC2
Service Profile vSphereBootTest1
Service Profile vSphereServBlade1
Service Profile vSphereServBlade2
Service Profile Templates
root
Policies
root
Pools
root
Server Pools
UUID Suffix Pools
New
Options
Exit
>> Servers › Service Profiles › root › Service Profile Oracle_RAC1
Service Profile Oracle_RAC1
General
Policies
vNICs
vHBAs
Boot Order
Server Details
FSM
Faults
Events
Fault Summary
0
0
0
0
Actions
Boot Server
Shutdown Server
Reset
KVM Console
SSH Console to SoL
Create a Clone
Create a Service Profile Template
Disassociate Service Profile
Change Service Profile Association
Unbind from the Template
Bind to a Template
Change UUID
Properties
Status
Status Details

Name: Oracle_RAC1
Description: Profile for Oracle RAC 11g UCS testing
UUID: 0
UUID Pool:
Associated Server: sys/chassis-2/blade-6
Service Profile Template:
Overall Status: ok

The operating system is not gracefully shut down but stopped instantaneously as shown in the circle in Figure 16. This type of shutdown emulates an external event that causes the server to suddenly fail.

Figure 16. Not Choosing Graceful Shutdown to Simulate a Hard Failure



Shutdown Service Profile Oracle_RAC1

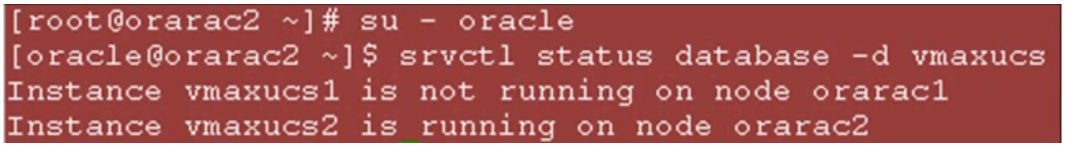
You are turning the server associated with this profile off.
To gracefully shut-down the server, OS needs to be shut-down.
Would you like me to shut-down the OS automatically?

□ Gracefully shutdown OS

If graceful OS shutdown is not selected or if it is not supported by the OS or it does not happen within reasonable amount of time, system will turn the server off non-gracefully.

After the server is down, the status of the Oracle RAC instance can be obtained with the same Oracle commands as before. This result in shown in Figure 17.

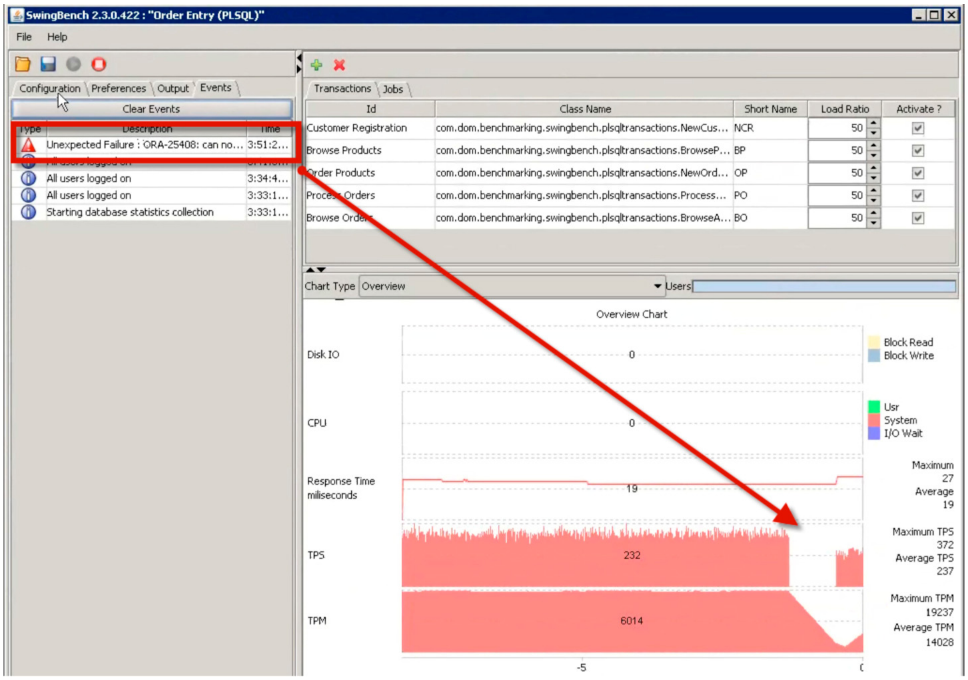
Figure 17. Oracle Shows Node Failure as Well



```
[root@orarac2 ~]# su - oracle
[oracle@orarac2 ~]$ srvctl status database -d vmaxucs
Instance vmaxucs1 is not running on node orarac1
Instance vmaxucs2 is running on node orarac2
```

Figure 18 shows the state of the workload generated from the Swingbench users during the blade-failure simulation. The figure shows that the loss of the compute node is accompanied by a significant decrease in activity due to the loss of users connected to the failed node. Nevertheless, the workload continues to run on the surviving node. This behavior validates the expected behavior of Oracle RAC.

Figure 18. Swingbench Shows Failure of Node and Drop in Throughput



SwingBench 2.3.0.422 : "Order Entry (PLSQL)"

Type	Description	Time
⚠	Unexpected Failure : ORA-25408: can no...	3:51:2...
ⓘ	All users logged on	[illegible]
ⓘ	All users logged on	3:34:4...
ⓘ	All users logged on	3:33:1...
ⓘ	Starting database statistics collection	3:33:1...

Id	Class Name	Short Name	Load Ratio	Activate ?
Customer Registration	com.dom.benchmarking.swingbench.plsqltransactions.NewCus...	NCR	50	☑
Browse Products	com.dom.benchmarking.swingbench.plsqltransactions.BrowseP...	BP	50	☑
Order Products	com.dom.benchmarking.swingbench.plsqltransactions.NewOrd...	OP	50	☑
Process Orders	com.dom.benchmarking.swingbench.plsqltransactions.Process...	PO	50	☑
Browse Orders	com.dom.benchmarking.swingbench.plsqltransactions.BrowseA...	BO	50	☑

Chart Type: Overview

Failure of Fabric Interconnect

A fabric interconnect is made to fail by unplugging one of the Cisco UCS 6120XP fabric interconnects. This action also simulates the loss of one of the LAN or SAN fabrics in the architecture.

Expected Outcomes

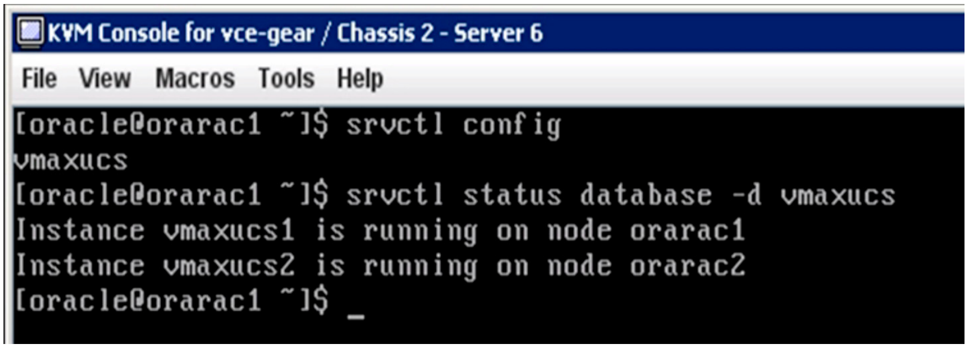
The failure of a fabric interconnect should result in the loss of paths to the devices accessed through the interconnect's uplink Fibre Channel ports. This behavior is no different than the failure of the Fibre Channel SAN component discussed earlier. The failure of the fabric interconnect also results in the transparent migration of the interblade and interchassis communication to the remaining fabric interconnect in the Cisco Unified Computing System. In addition to the aforementioned behavior, a gratuitous Address Resolution Protocol (ARP) is sent by the surviving fabric interconnect to the upstream IP switch to register the MAC addresses associated to the failed interconnect.

The behavior described here should manifest itself in different ways for the applications running on the Cisco Unified Computing System. The failure of the Fibre Channel paths should be recognized by PowerPath and result in automatic redirection of the I/O processing to the surviving path. In addition, the IP traffic (both external and internode) should not be affected by the fabric interconnect failure.

Results

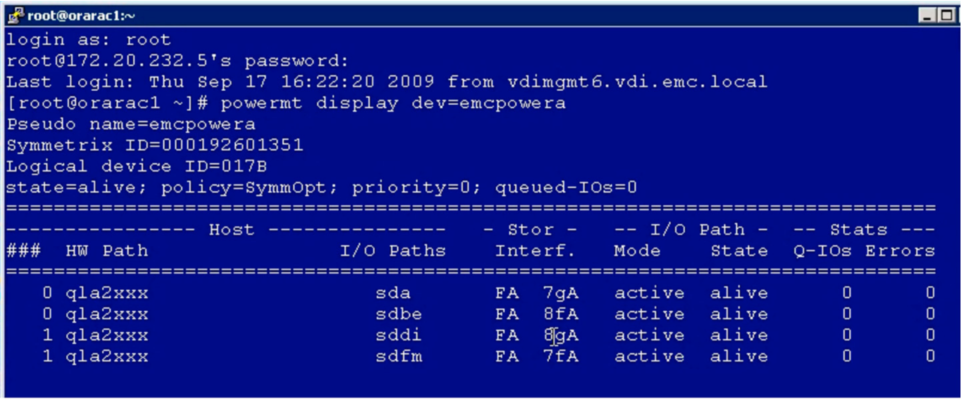
Figures 19 and 20 show the nodes during normal operation. As the figures show, the Oracle RAC instance is running on two nodes, and all four paths to the devices are active and handling I/O requests.

Figure 19. Normal Operation



```
[oracle@orarac1 ~]$ srvctl config
vmaxucs
[oracle@orarac1 ~]$ srvctl status database -d vmaxucs
Instance vmaxucs1 is running on node orarac1
Instance vmaxucs2 is running on node orarac2
[oracle@orarac1 ~]$ _
```

Figure 20. EMC PowerPath CLI Output Showing Healthy Paths

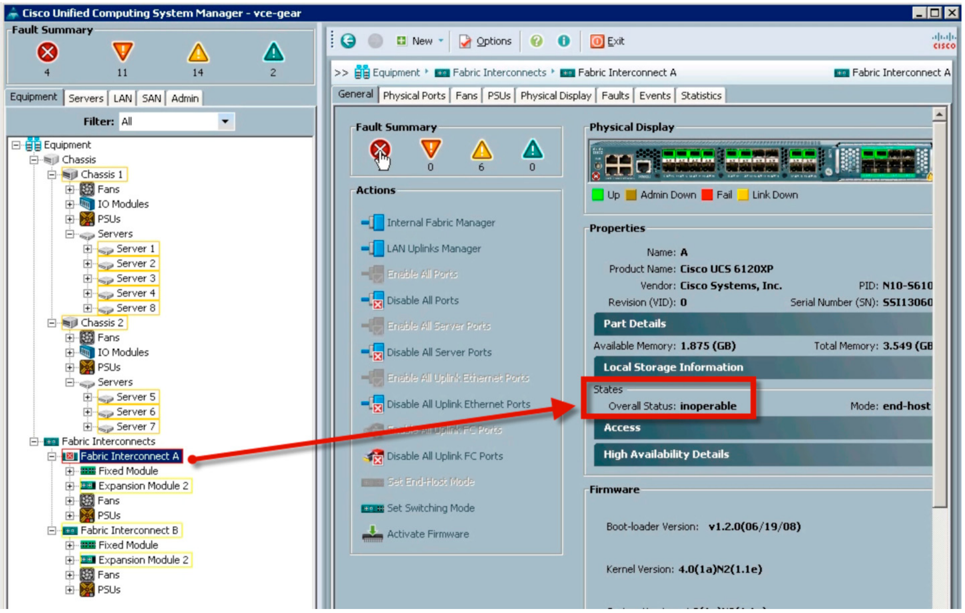


```
login as: root
root@172.20.232.5's password:
Last login: Thu Sep 17 16:22:20 2009 from vdimgmt6.vdi.emc.local
[root@orarac1 ~]# powermt display dev=emcpowera
Pseudo name=emcpowera
Symmetrix ID=000192601351
Logical device ID=017B
state=alive; policy=SymmOpt; priority=0; queued-IOs=0
==============================================================================
---------------- Host ---------------   - Stor -   -- I/O Path -  -- Stats ---
###  HW Path                I/O Paths    Interf.   Mode    State   Q-IOs Errors
==============================================================================
   0 qla2xxx                   sda       FA  7gA   active  alive       0      0
   0 qla2xxx                   sdbe      FA  8fA   active  alive       0      0
   1 qla2xxx                   sddi      FA  8gA   active  alive       0      0
   1 qla2xxx                   sdfm      FA  7fA   active  alive       0      0
```

The failure of the fabric interconnect is injected by pulling the power cord on the back of the fabric interconnect supporting “Fabric A”. This method was used primarily because a graceful shutdown, if available, does not simulate the problems that can be exposed during a catastrophic failure of the component.

The Cisco UCS Manager screens in Figures 21 show the power failure and the red X indicating that the interconnect is offline.

Figure 21. Cisco UCS Manager Showing That the Interconnect Is Down



Name: A
Product Name: Cisco UCS 6120XP
Vendor: Cisco Systems, Inc.
PID: N10-S610
Revision (VID): 0
Serial Number (SN): SSI13060
Available Memory: 1.875 (GB)
Total Memory: 3.549 (GB
Overall Status: inoperable
Mode: end-host
Boot-loader Version: v1.2.0(06/19/08)
Kernel Version: 4.0(1a)N2(1.1e)

The workload generated using Swingbench also shows the brief slowdown in I/O activity as the PowerPath multipathing software detects the lost paths and reroutes all I/O processing to the remaining paths (see Figures 22 and 23). This behavior and error state propagation is identical in nature to the Fibre Channel uplink failure discussed earlier. The behavior shown in the figures validates the innate HA architecture of the Cisco Unified Computing System with its two redundant fabric interconnects.

Note that the external users simulated by Swingbench experienced no disconnects as the TCP connection was transparently moved to the surviving fabric interconnect. Furthermore, although not shown, the cluster heartbeat which is routed through the fabric interconnects is not lost during the failure event.

The fabric interconnects replicate state data between them on a transaction basis through the cluster interconnects. I/O processing is active on both fabrics, and the fabric extenders are redundant, which is an integral part of the Cisco Unified Computing System architecture.

Figure 22. Swingbench GUI Output Showing Drop in Throughput

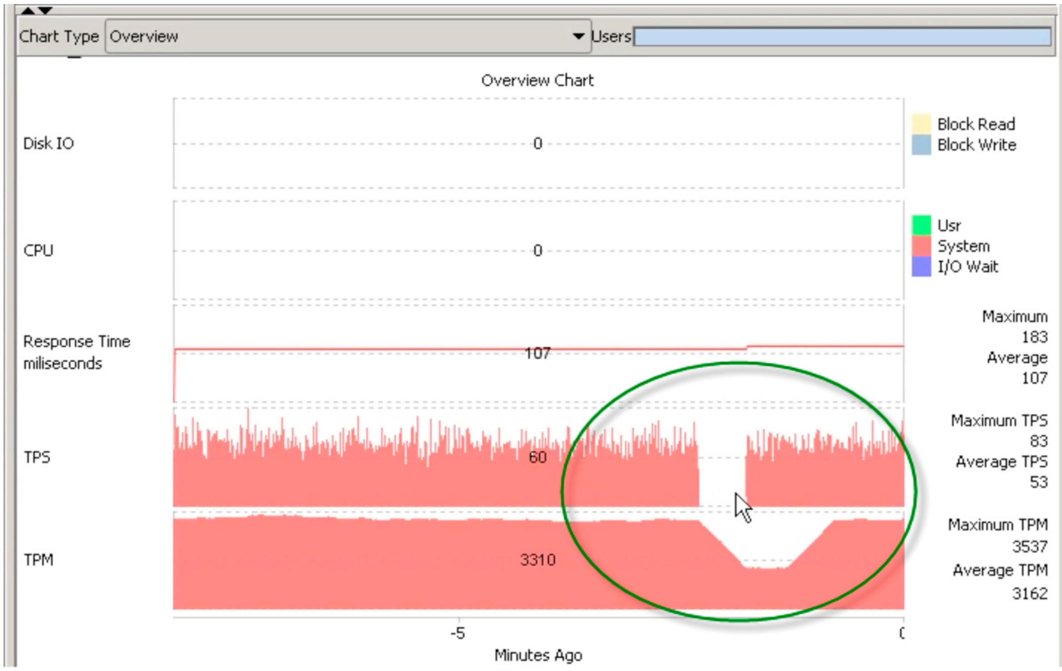
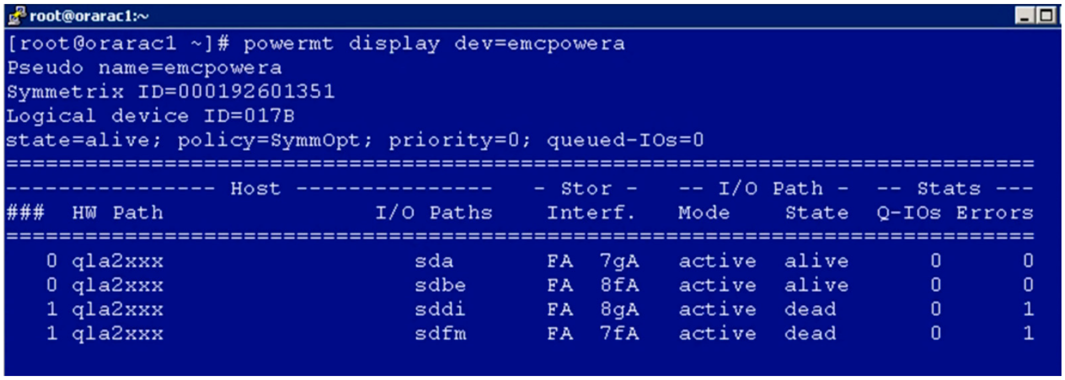


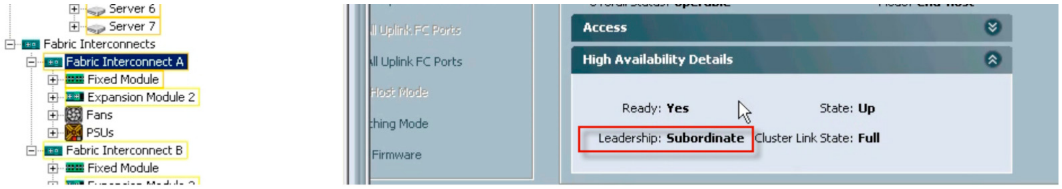
Figure 23. EMC PowerPath Showing That Paths Are Down After Fabric Interconnect Failure



```
[root@orarac1 ~]# powermt display dev=emcpowera
Pseudo name=emcpowera
Symmetrix ID=000192601351
Logical device ID=017B
state=alive; policy=SymmOpt; priority=0; queued-IOs=0
==============================================================================
---------------- Host ---------------   - Stor -   -- I/O Path -   -- Stats ---
###  HW Path                I/O Paths    Interf.   Mode    State   Q-IOs Errors
==============================================================================
  0 qla2xxx                   sda       FA  7gA   active  alive      0      0
  0 qla2xxx                   sdbe      FA  8fA   active  alive      0      0
  1 qla2xxx                   sddi      FA  8gA   active  dead       0      1
  1 qla2xxx                   sdfm      FA  7fA   active  dead       0      1
```

When the power is restored to the fabric interconnect, the cluster state between the two fabric interconnects is automatically re-established, all paths of the devices are restored, and normal operations resume. After power is restored to the fabric interconnect, no user administration is required to initiate the reformation of the high-availability cluster between the fabric interconnects, as shown in Figure 24. The figure shows that Fabric Interconnect A has been restored as subordinate to Fabric Interconnect B because Fabric Interconnect B assumed the primary role when Fabric Interconnect A failed.

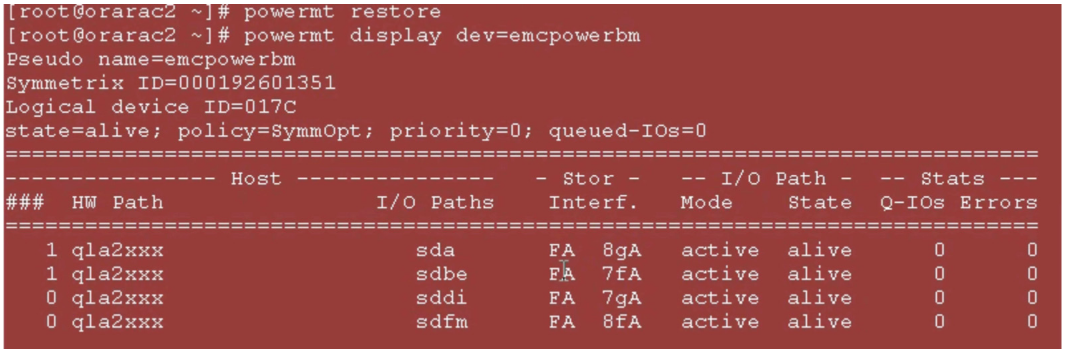
Figure 24. Cisco Unified Computing System Showing That Fabric A Is Back But in a Subordinate Role



Ready: **Yes**
State: **Up**
Leadership: **Subordinate**
Cluster Link State: **Full**

Figure 25 shows that the PowerPath software is easily able to restore the failed paths after the interconnect comes back online.

Figure 25. EMC PowerPath CLI Showing That All Paths Are Online



```
[root@orarac2 ~]# powermt restore
[root@orarac2 ~]# powermt display dev=emcpowerbm
Pseudo name=emcpowerbm
Symmetrix ID=000192601351
Logical device ID=017C
state=alive; policy=SymmOpt; priority=0; queued-IOs=0
==============================================================================
--------------- Host ---------------   - Stor -   -- I/O Path -  -- Stats ---
###  HW Path                I/O Paths    Interf.   Mode    State  Q-IOs Errors
==============================================================================
   1 qla2xxx                   sda       FA  8gA   active  alive      0      0
   1 qla2xxx                   sdbe      FA  7fA   active  alive      0      0
   0 qla2xxx                   sddi      FA  7gA   active  alive      0      0
   0 qla2xxx                   sdfm      FA  8fA   active  alive      0      0
```

Scaling Oracle RAC by Adding Nodes

The goal of this test was to show how easily another node can be added to an already running RAC instance with EMC disk cloning and the use of Cisco Unified Computing System service profile templates and resource pools (server pools, WWPN, UUID, and MAC addresses). The main use case addresses the elasticity requirements of cloud architectures, which call for the easy addition of compute resources to satisfy increased demand from an application.

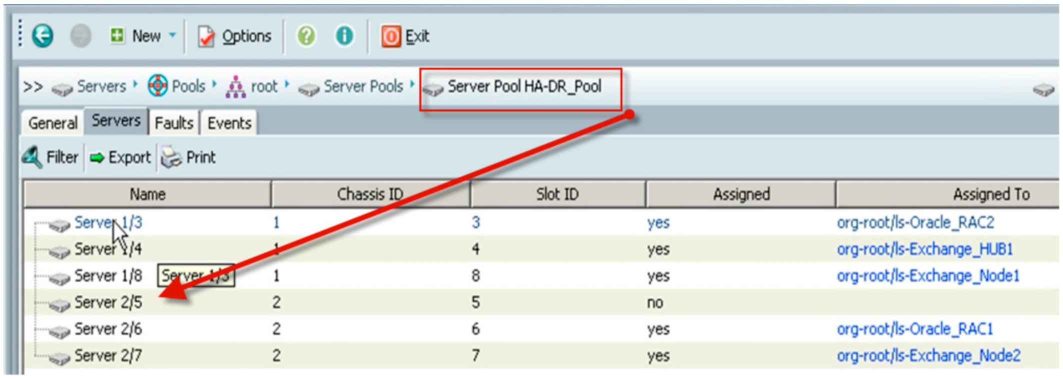
To provision a new node, the following tasks must be performed:

- New compute blade must be identified and configured properly for storage and network access, including configuration of SAN zoning and masking for access to the OS and data LUNs and Ethernet connectivity
- Firmware must be installed on the blade to help ensure compatibility
- The OS and Oracle RAC binary images must be installed and configured on this node
- The node must be added to the cluster using Oracle administration utilities so that it can properly join the cluster

With the joint EMC and Cisco solution, these steps can be highly simplified and automated with a few GUI clicks using the innate architecture of the Cisco Unified Computing System and the VMAX storage array. The process can also be further automated using simple scripting languages.

The Cisco Unified Computing System server pool construct is used along with service profile templates to create another copy of an existing service profile, select a free server from the designated server pool, and assign unique ID values to all the necessary attributes in the service profile. Figure 26 shows the spare server in the existing server pool. This server will be used here to show the simplicity of the process for adding compute nodes to a RAC instance.

Figure 26. Server Pool and Service Profile Assignments

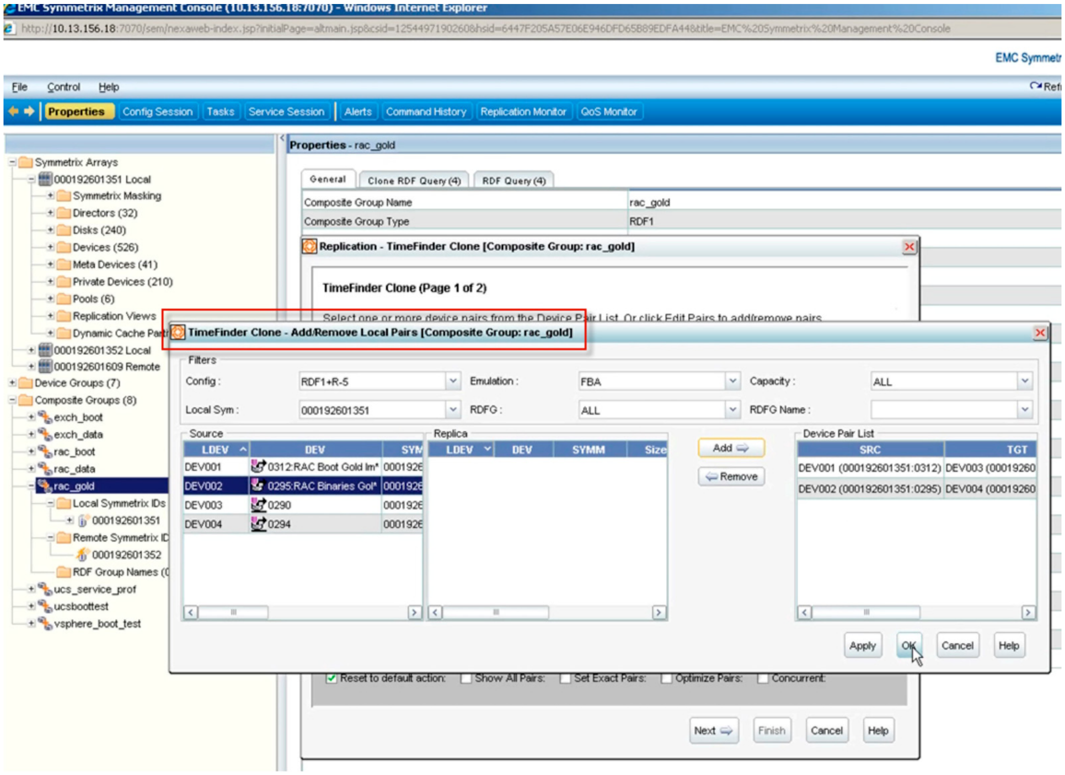


Name	Chassis ID	Slot ID	Assigned	Assigned To
Server 1/3	1	3	yes	org-root/ls-Oracle_RAC2
Server 1/4	1	4	yes	org-root/ls-Exchange_HUB1
Server 1/8	1	8	yes	org-root/ls-Exchange_Node1
Server 2/5	2	5	no	
Server 2/6	2	6	yes	org-root/ls-Oracle_RAC1
Server 2/7	2	7	yes	org-root/ls-Exchange_Node2

The first step in the process is to use EMC TimeFinder software to clone a master (or gold) image of the operating system. The gold image of the operating system is similar to the image for the other RAC nodes, but it includes specialized scripts and parameters that allow customization of the operating system when the image is booted for the first time.

Figure 27 shows the management software for cloning the operating system image from a gold-copy template of RAC node using EMC TimeFinder.

Figure 27. EMC TimeFinder Clone Operation GUI



Composite Group Name	rac_gold
Composite Group Type	RDF1

Replication - TimeFinder Clone [Composite Group: rac_gold]

TimeFinder Clone (Page 1 of 2)

Select one or more device pairs from the Device Pair List. Or click Edit Pairs to add/remove pairs

TimeFinder Clone - Add/Remove Local Pairs [Composite Group: rac_gold]

Filters

Config :	RDF1+R-5	Emulation :	FBA	Capacity :	ALL
Local Sym :	000192601351	RDFG :	ALL	RDFG Name :	

Source

LDEV	DEV	SYM
DEV001	0312:RAC Boot Gold Im*	0001926
DEV002	0295:RAC Binaries Gol*	0001926
DEV003	0290	0001926
DEV004	0294	0001926

Replica

LDEV	DEV	SYMM	Size

Device Pair List

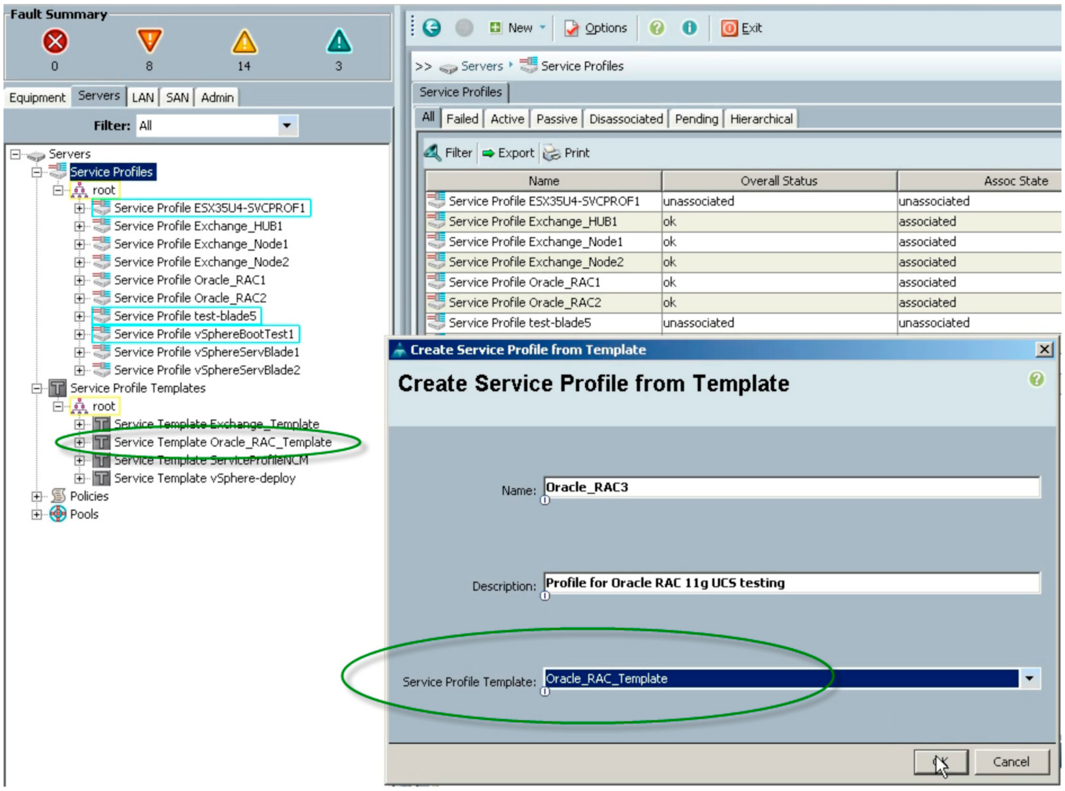
SRC	TGT
DEV001 (000192601351:0312)	DEV003 (00019260
DEV002 (000192601351:0295)	DEV004 (00019260

☑ Reset to default action: ☐ Show All Pairs: ☐ Set Exact Pairs: ☐ Optimize Pairs: ☐ Concurrent:

The Cisco UCS Manager automatically assigns the WWPN for the blades from a predefined pool. This capability of Cisco UCS Manager enables assignment of the boot volume to the additional RAC node ahead of time. This assignment is achieved by proactively creating the appropriate initiator groups, port groups, and storage group using either the Symmetrix Management Console (SMC) or Solutions Enabler CLI (SYMCLI).

The next step in the process is to use Cisco UCS Manager to create a new service profile for the new RAC node from an existing RAC service profile template, as shown in Figure 28.

Figure 28. Creating a Service Profile from an Existing Template in Cisco Unified Computing System


Fault Summary
0
8
14
3
Equipment
Servers
LAN
SAN
Admin
Filter: All
Servers
Service Profiles
root
Service Profile ESX35U4-SVCPROF1
Service Profile Exchange_HUB1
Service Profile Exchange_Node1
Service Profile Exchange_Node2
Service Profile Oracle_RAC1
Service Profile Oracle_RAC2
Service Profile test-blade5
Service Profile vSphereBootTest1
Service Profile vSphereServBlade1
Service Profile vSphereServBlade2
Service Profile Templates
root
Service Template Exchange_Template
Service Template Oracle_RAC_Template
Service Template ServiceProfileNCM
Service Template vSphere-deploy
Policies
Pools
New
Options
Exit
>> Servers › Service Profiles
Service Profiles
All
Failed
Active
Passive
Disassociated
Pending
Hierarchical
Filter
Export
Print

Name	Overall Status	Assoc State
Service Profile ESX35U4-SVCPROF1	unassociated	unassociated
Service Profile Exchange_HUB1	ok	associated
Service Profile Exchange_Node1	ok	associated
Service Profile Exchange_Node2	ok	associated
Service Profile Oracle_RAC1	ok	associated
Service Profile Oracle_RAC2	ok	associated
Service Profile test-blade5	unassociated	unassociated

Create Service Profile from Template

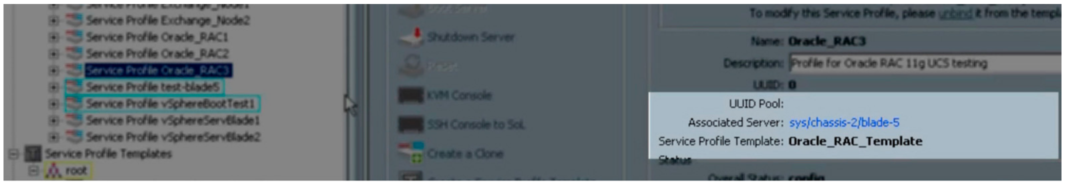
Name: Oracle_RAC3

Description: Profile for Oracle RAC 11g UCS testing

Service Profile Template: Oracle_RAC_Template

Figure 29 shows the results from the previous operation. The figure shows that the service profile has been created and automatically assigned from the server pool to the free blade. This process results in a blade that is fully ready to use for RAC.

Figure 29. Association of Service Profiles with Blades


Service Profile Exchange_Node2
Service Profile Oracle_RAC1
Service Profile Oracle_RAC2
Service Profile Oracle_RAC3
Service Profile test-blade5
Service Profile vSphereBootTest1
Service Profile vSphereServBlade1
Service Profile vSphereServBlade2
Service Profile Templates
root
Shutdown Server
Reset
KVM Console
SSH Console to SoL
Create a Clone
Status

To modify this Service Profile, please unbind it from the templ

Name: Oracle_RAC3

Description: Profile for Oracle RAC 11g UCS testing

UUID: 0

UUID Pool:

Associated Server: sys/chassis-2/blade-5

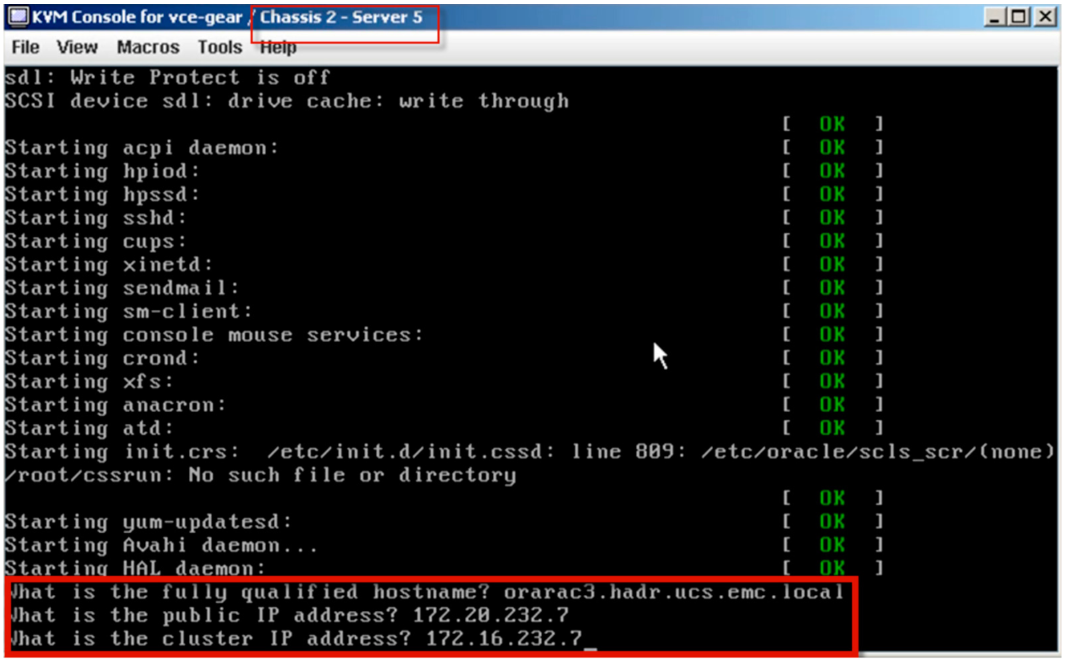
Service Profile Template: Oracle_RAC_Template

Overall Status: config

The blade automatically reboots with the new attributes using the cloned operating system image assigned through the provisioning steps cited in the previous paragraphs. The zoning is already configured on the SAN switch, and the boot policy is assigned to the service profile template.

The assignment of the host name, IP addresses and the DNS servers for the additional RAC node can be automated by creating a script that is executed once during the initial boot of the node. This script and the parameter controlling its execution is included in the gold image from which the boot disk for the additional RAC node was cloned. A sample of the script used in this test is shown in Appendix A. Figure 30 shows the booting of the new blade as seen from the keyboard, video, and mouse (KVM) console; it shows the execution of the script during the boot of the additional node used in this test.

Figure 30. Example of Preboot Script Output

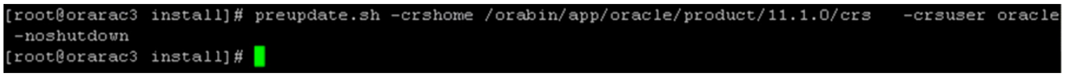

KVM Console for vce-gear / Chassis 2 - Server 5
File View Macros Tools Help

```
sdl: Write Protect is off
SCSI device sdl: drive cache: write through
                                                           [  OK  ]
Starting acpi daemon:                                      [  OK  ]
Starting hpiod:                                            [  OK  ]
Starting hpssd:                                            [  OK  ]
Starting sshd:                                             [  OK  ]
Starting cups:                                             [  OK  ]
Starting xinetd:                                           [  OK  ]
Starting sendmail:                                         [  OK  ]
Starting sm-client:                                        [  OK  ]
Starting console mouse services:                           [  OK  ]
Starting crond:                                            [  OK  ]
Starting xfs:                                              [  OK  ]
Starting anacron:                                          [  OK  ]
Starting atd:                                              [  OK  ]
Starting init.crs:  /etc/init.d/init.cssd: line 809: /etc/oracle/scls_scr/(none)
/root/cssrun: No such file or directory
                                                           [  OK  ]
Starting yum-updatesd:                                     [  OK  ]
Starting Avahi daemon...                                   [  OK  ]
Starting HAL daemon:                                       [  OK  ]
What is the fully qualified hostname? orarac3.hadr.ucs.emc.local
What is the public IP address? 172.20.232.7
What is the cluster IP address? 172.16.232.7_
```

After the node is fully booted, cloning scripts specific to Oracle RAC must be executed to allow the node to join the RAC instance. The complete procedure used in this test is provided in Appendix A. However, to provide an overview of the flow of operations, the following figures show the execution of the scripts.

Because the boot disk contained all the Oracle binaries, the cloning scripts to add the third RAC node can be run as soon as the new node boots. The process begins by updating the third node's Oracle CRS home files using the **preupdate.sh** script, which prepares the home file for the **clone.pl** script (Figure 31).

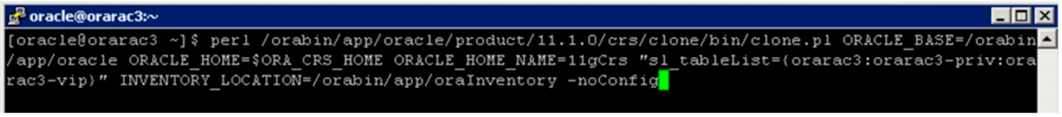
Figure 31. The **preupdate.sh** Oracle Clusterware Script



```
[root@orarac3 install]# preupdate.sh -crshome /orabin/app/oracle/product/11.1.0/crs   -crsuser oracle
 -noshutdown
[root@orarac3 install]# 
```

Next, the Oracle CRS home file can be cloned with the **clone.pl** script. This script will configure the Oracle CRS home file for this new node, providing the necessary parameters for it to join the cluster (Figure 32).

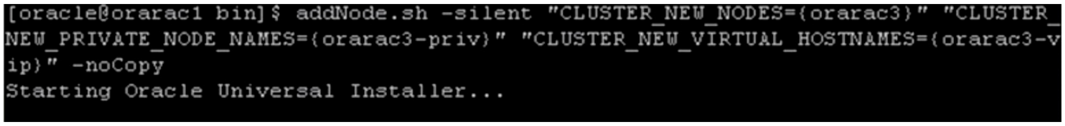
Figure 32. The **clone.pl** Script



```
[oracle@orarac3 ~]$ perl /orabin/app/oracle/product/11.1.0/crs/clone/bin/clone.pl ORACLE_BASE=/orabin
/app/oracle ORACLE_HOME=$ORA_CRS_HOME ORACLE_HOME_NAME=11gCrs "sl_tableList={orarac3:orarac3-priv:ora
rac3-vip}" INVENTORY_LOCATION=/orabin/app/oraInventory -noConfig
```

Finally, the **addNode.sh** script is run, which sets the third node to join the other two nodes as part of the RAC instance (Figure 33).

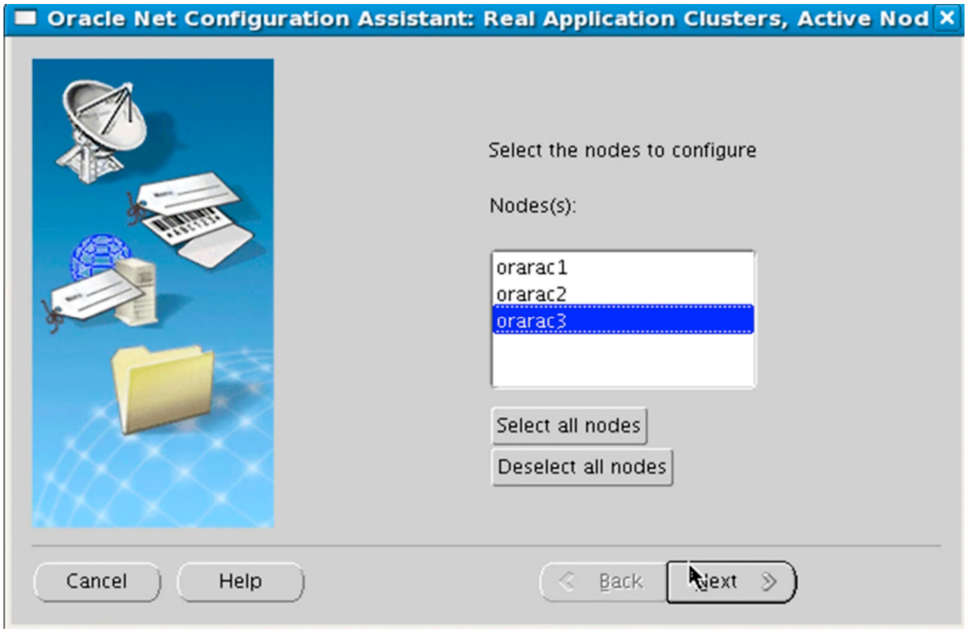
Figure 33. The **addNode.sh** Script



```
[oracle@orarac1 bin]$ addNode.sh -silent "CLUSTER_NEW_NODES={orarac3}" "CLUSTER_
NEW_PRIVATE_NODE_NAMES={orarac3-priv}" "CLUSTER_NEW_VIRTUAL_HOSTNAMES={orarac3-v
ip}" -noCopy
Starting Oracle Universal Installer...
```

The new node is now part of the foundation of the RAC instance: that is, the cluster has been extended. However to complete the process, both Oracle Automatic Storage Management (ASM) and the database need extending. This is accomplished using Oracle's GUI tools, the first of which is the Oracle Net Configuration Assistant, which is used to add the database listener for the third node (Figure 34).

Figure 34. Oracle Net Configuration Assistant



Oracle Net Configuration Assistant: Real Application Clusters, Active Nod

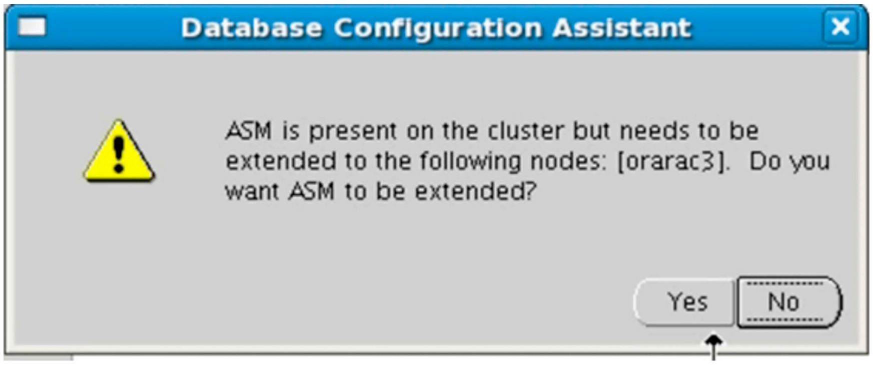
Select the nodes to configure

Nodes(s):

orarac1
orarac2
orarac3

Next, the Oracle GUI's Database Configuration Assistant (DBCA) is used to add both the Oracle ASM instance and the database instance. Both additions are simple point-and-click operations; Figure 35 shows the addition of Oracle ASM.

Figure 35. DBCA

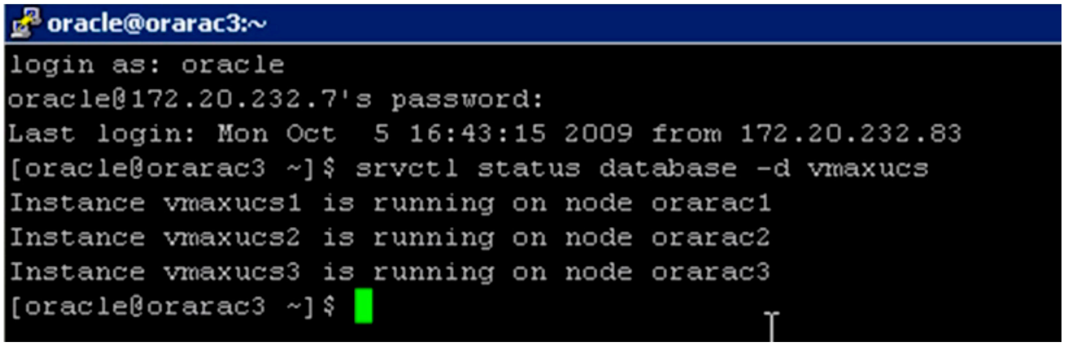


Database Configuration Assistant

ASM is present on the cluster but needs to be extended to the following nodes: [orarac3]. Do you want ASM to be extended?

Finally, after completion of the DBCA utility, you can run the Oracle server control CLI, **srvctl**, to see the status of all the instances in the cluster and confirm that the addition of the third RAC node is complete, and thus that the scaling succeeded (Figure 36).

Figure 36. The srvctl Status Listing



```
login as: oracle
oracle@172.20.232.7's password:
Last login: Mon Oct  5 16:43:15 2009 from 172.20.232.83
[oracle@orarac3 ~]$ srvctl status database -d vmaxucs
Instance vmaxucs1 is running on node orarac1
Instance vmaxucs2 is running on node orarac2
Instance vmaxucs3 is running on node orarac3
[oracle@orarac3 ~]$
```

Part 2: Disaster Recovery from Site Failures

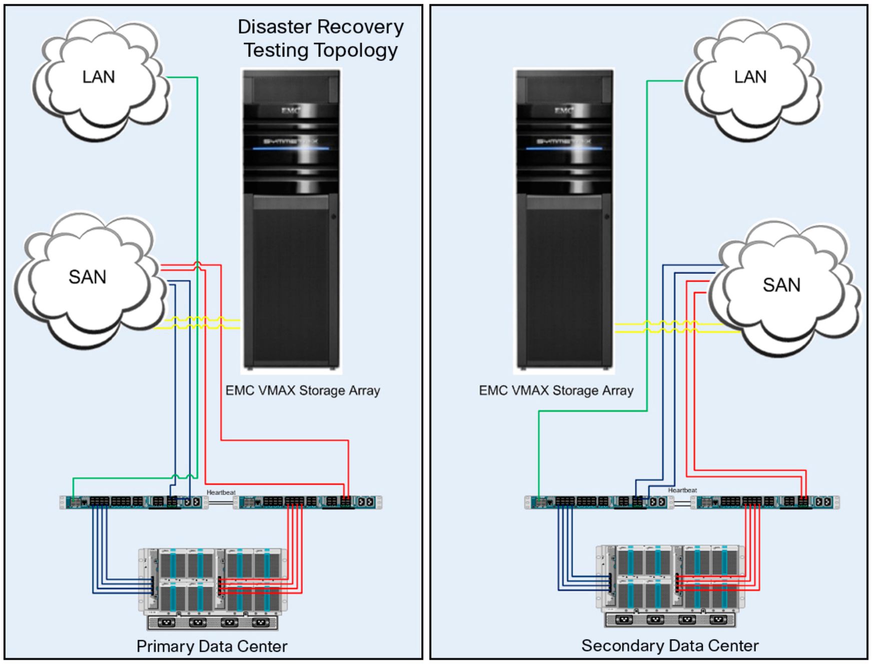
Description, Topology, and Assumptions

Figure 37 shows the topology of the architecture to describe the process and procedures for recovering from a site failure with little effort. The setup simulates a common configuration for customers who have multiple buildings in close proximity that can support synchronous data replication.

The test assumed a flat network between the two sites used in the configuration. It also assumed that the same networks, VLANs, and VSANs exist in each Cisco Unified Computing System domain. These assumptions are consistent with most campus and metropolitan area network (MAN) deployments.

The test used two independent and active Cisco Unified Computing System domains (D1, D2) in a steady-state solution prior to the simulation of a disaster. D1 was the production domain, and D2 was the test and development domain (Figure 37).

Figure 37. Overall Disaster Recovery Topology: Two Separate Cisco Unified Computing System Domains (Domain 1 and Domain 2) with Synchronous Replication to Two Different EMC Symmetrix VMAX Systems

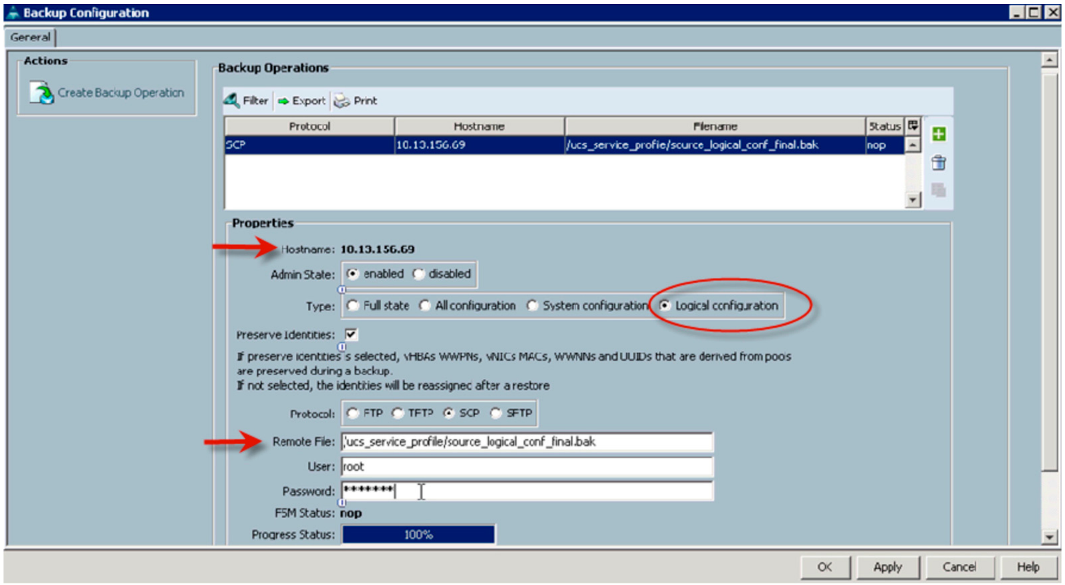


An earlier section discussed the solution at a logical level and provided a solution overview. This section steps through the normal operating environment and shows the flow of actions to bring the application online in the second Cisco Unified Computing System domain after a site failure. A complete building failure is simulated by powering off both Cisco UCS 6120XP fabric interconnects associated with D1 of the topology. The normal operating environment for the test is a three-node RAC cluster running within D1.

The first step in providing disaster restart protection is the replication of all of the necessary devices from the production site to the disaster restart site. If the business process spans multiple application domains and servers, EMC Consistency Group technology should be used to help ensure that the data replicated between the two sites preserves business-process consistency.

The next step in the process is creating a backup of the Cisco Unified Computing System configuration at D1 using the Cisco UCS Manager (Figure 38).

Figure 38. Cisco UCS Manager Backup Operation



Protocol	Hostname	Filename	Status
SCP	10.13.156.69	/ucs_service_profie/scurce_logical_conf_final.bak	nop

Properties

Hostname: 10.13.156.69

Admin State: ● enabled ○ disabled

Type: ○ Full state ○ All configuration ○ System configuration ● Logical configuration

Preserve Identities: ☑

If preserve identities is selected, vHBAs WWPNs, vNICs MACs, WWNNs and UUIDs that are derived from pools are preserved during a backup.
If not selected, the identities will be reassigned after a restore

Protocol: ○ FTP ○ TFTP ● SCP ○ SFTP

Remote File: /ucs_service_profile/source_logical_conf_final.bak

User: root

Password: *******

FSM Status: nop

Progress Status: 100%

As the figure shows, the IP address of a remote host is specified along with a valid path on that host. In this test, a Linux server was used for this purpose.

A full-state binary image backup is not appropriate in the scenario considered in this test because the second Cisco Unified Computing System domain is active and is running other applications. A full-state backup would require replacement of the configuration of Cisco Unified Computing System D2, which is not the goal here. Rather, the goal is to add the service profiles, pools, and policies to D2 and bring the blades online and start executing the applications that have failed at the production site. This can be achieved by creating a logical backup that contains all of the service profile information that is needed.

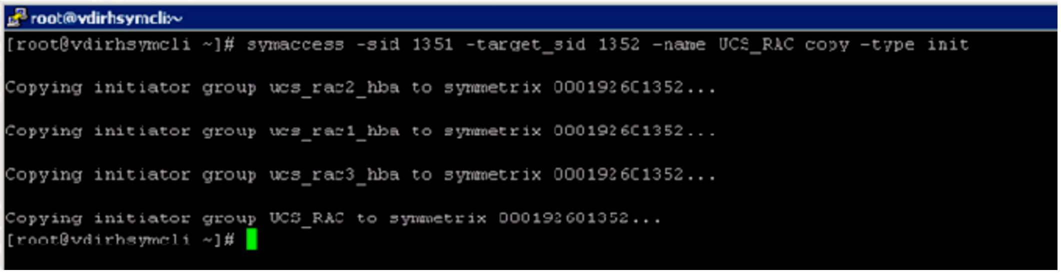
You can perform one or more of the following types of backup operations using the Cisco Unified Computing System

- Full state: Includes a snapshot of the entire system. You can use the file generated from this backup to restore the system during disaster recovery. This file can restore or rebuild the configuration on the original fabric interconnect or re-create the configuration on a different fabric interconnect. You cannot use this file for an import operation.
- All configurations: Includes all system and logical configuration settings. You can use the file generated from this backup to import these configuration settings to the original fabric interconnect or to a different fabric interconnect. You cannot use this file for a system restore operation.

- System configuration: Includes all system configuration settings such as usernames, roles, and locales. You can use the file generated from this backup to import these configuration settings to the original fabric interconnect or to a different fabric interconnect. You cannot use this file for a system restore operation.
- Logical configuration: Includes all logical configuration settings such as service profiles, VLANs, VSANs, pools, and policies. You can use the file generated from this backup to import these configuration settings to the original fabric interconnect or to a different fabric interconnect. You cannot use this file for a system restore operation.

The backup file, in XML format, exists on the server that was specified in the backup operation screen. Figure 39 shows the backup file on the server used in this test. The backup of the Cisco Unified Computing System configuration should be performed frequently and any time the configuration is changed. This backup operation is necessary to help ensure that all pertinent information is captured and can be restored at the disaster recovery site in case of a failure.

Figure 39. Output Showing XML File



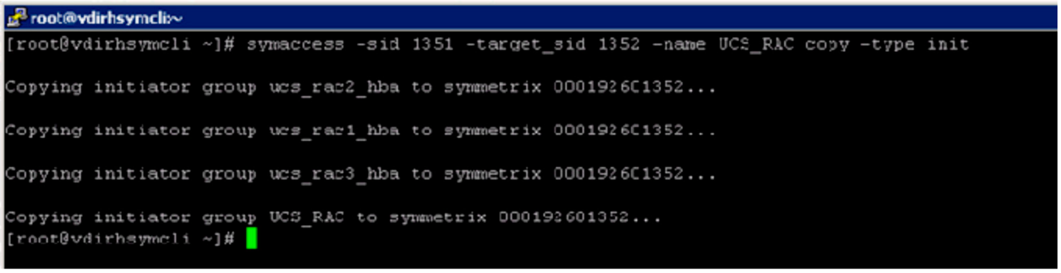
```
[root@vdirhsymcli ~]# symaccess -sid 1351 -target_sid 1352 -name UCS_RAC copy -type init

Copying initiator group ucs_rac2_hba to symmetrix 0001926C1352...

Copying initiator group ucs_rac1_hba to symmetrix 0001926C1352...

Copying initiator group ucs_rac3_hba to symmetrix 0001926C1352...

Copying initiator group UCS_RAC to symmetrix 000192601352...
[root@vdirhsymcli ~]#
```

The solution presented in this document re-creates the production environment at the disaster recovery site, including the WWPNs and MAC addresses associated to the servers. Therefore, the next step in the solution is the creation of the zoning and appropriate LUN masking on the SAN fabric and VMAX array, respectively, at the disaster recovery site.

EMC Solutions Enabler Version 7.1 introduces a new feature to simplify the creation of the appropriate LUN masking entries on the VMAX array at the disaster recovery site. This unique feature is very compelling when combined with Cisco Unified Computing System service profile mobility. Using both of these technologies together enables the service profiles to come online very easily on the remote site.

Figure 40 shows the command to copy the initiator groups that contain the WWPNs of the host bus adapters (HBAs) of the Cisco Unified Computing System nodes at the production site from the VMAX array at the production site to the VMAX array at the disaster recovery site. The use of this feature allows a large environment to be easily re-created with little effort and without errors on the VMAX array at the disaster recovery site. Similar commands exist to copy port group and storage group definitions and views.

Figure 40. The **symaccess** CLI Command Output



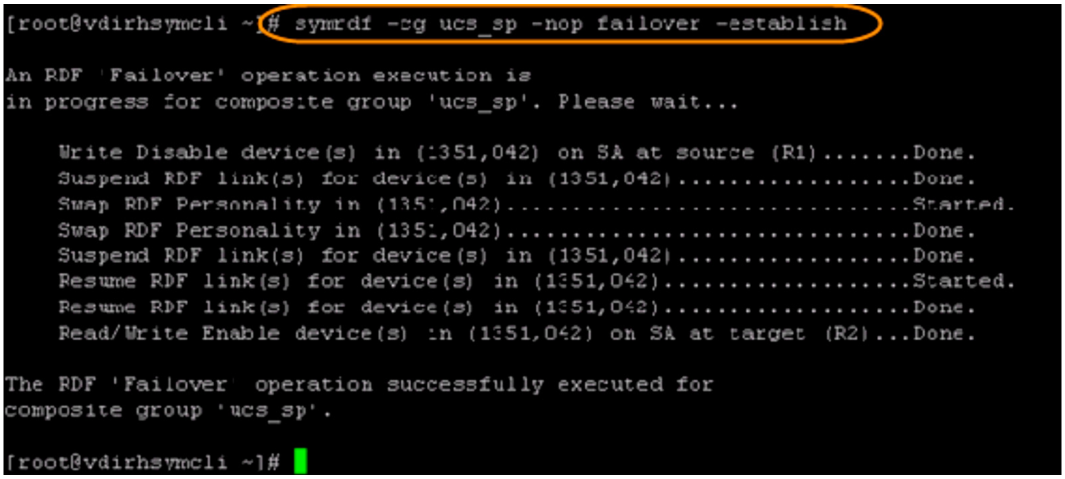
```
[root@vdirhsymcli ~]# symaccess -sid 1351 -target_sid 1352 -name UCS_RAC copy -type init

Copying initiator group ucs_rac2_hba to symmetrix 0001926C1352...

Copying initiator group ucs_rac1_hba to symmetrix 0001926C1352...

Copying initiator group ucs_rac3_hba to symmetrix 0001926C1352...

Copying initiator group UCS_RAC to symmetrix 000192601352...
[root@vdirhsymcli ~]#
```

Following the steps just described helps ensure a clean and simple recovery in the event of a disaster in the production environment.

Recovery Procedures After a Disaster

The first step in the recovery process is to use the Symmetrix management software (Symmetrix Management Console or Solutions Enabler CLI) to read/write-enable the devices at the disaster recovery site. Furthermore, depending on the type of failure, to dynamically swap the personalities of the Symmetrix devices to start replication of the data from the disaster recovery site back to the production site. Figure 41 shows the command to read/write-enable all of the devices associated with the solution on the VMAX array at the disaster recovery site. The figure also shows that the direction of the replication is reversed.

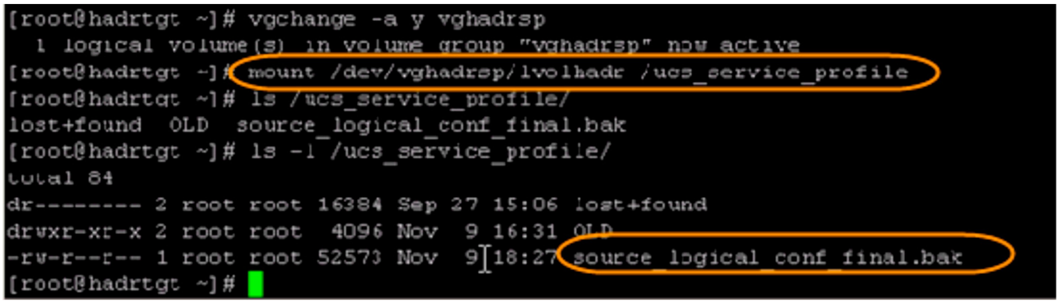
Figure 41. The **symrdf** Failover Command Output



```
[root@vdirhsymcli ~]# symrdf -cg ucs_sp -nop failover -establish

An RDF 'Failover' operation execution is
in progress for composite group 'ucs_sp'. Please wait...

    Write Disable device(s) in (1351,042) on SA at source (R1).......Done.
    Suspend RDF link(s) for device(s) in (1351,042).................Done.
    Swap RDF Personality in (1351,042)..............................Started.
    Swap RDF Personality in (1351,042)..............................Done.
    Suspend RDF link(s) for device(s) in (1351,042).................Done.
    Resume RDF link(s) for device(s) in (1351,042)..................Started.
    Resume RDF link(s) for device(s) in (1351,042)..................Done.
    Read/Write Enable device(s) in (1351,042) on SA at target (R2)...Done.

The RDF 'Failover' operation successfully executed for
composite group 'ucs_sp'.

[root@vdirhsymcli ~]#
```

As mentioned in an earlier section and depicted in Figure 37, the file system containing the logical backup of Cisco Unified Computing System domain D1 is replicated using SRDF. After the remote devices are enabled for reading and writing, the file system containing the backups can be mounted on a Linux server at the remote site, as shown in Figure 42.

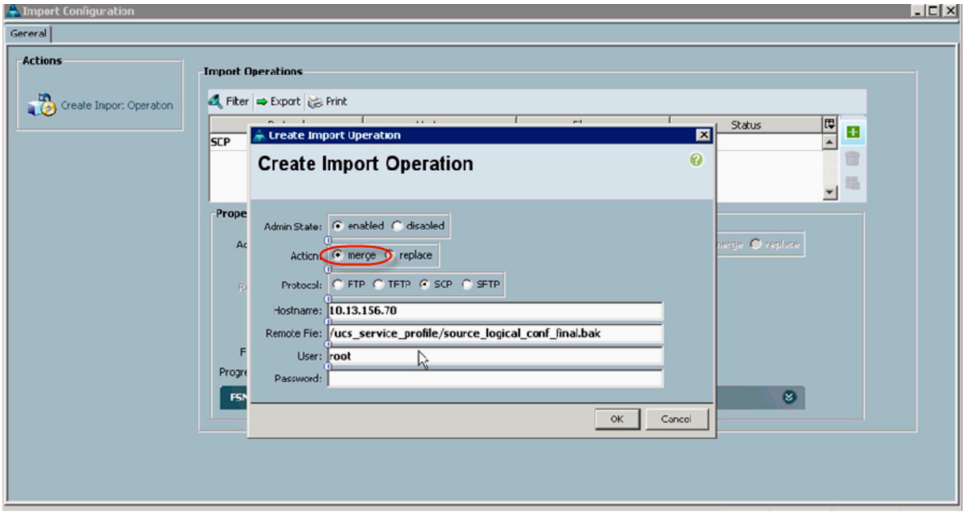
Figure 42. Changing LVM Volume Group to Active and Mounting the File System



```
[root@hadrtgt ~]# vgchange -a y vghadrsp
  1 logical volume(s) in volume group "vghadrsp" now active
[root@hadrtgt ~]# mount /dev/vghadrsp/lvolhadr /ucs_service_profile
[root@hadrtgt ~]# ls /ucs_service_profile/
lost+found  OLD  source_logical_conf_final.bak
[root@hadrtgt ~]# ls -l /ucs_service_profile/
total 64
dr-------- 2 root root 16384 Sep 27 15:06 lost+found
drwxr-xr-x 2 root root  4096 Nov  9 16:31 OLD
-rw-r--r-- 1 root root 52573 Nov  9 18:27 source_logical_conf_final.bak
[root@hadrtgt ~]#
```

The next step in the process is to log on to Cisco UCS Manager for D2 and use the Import Configuration utility to import the existing backup file, as shown in Figure 43.

Figure 43. Cisco UCS Manager Import Operation



Create Import Operation

Admin State: ● enabled ○ disabled

Action: ● merge ○ replace

Protocol: ○ FTP ○ TFTP ● SCP ○ SFTP

Hostname: 10.13.156.70

Remote File: /ucs_service_profile/source_logical_conf_final.bak

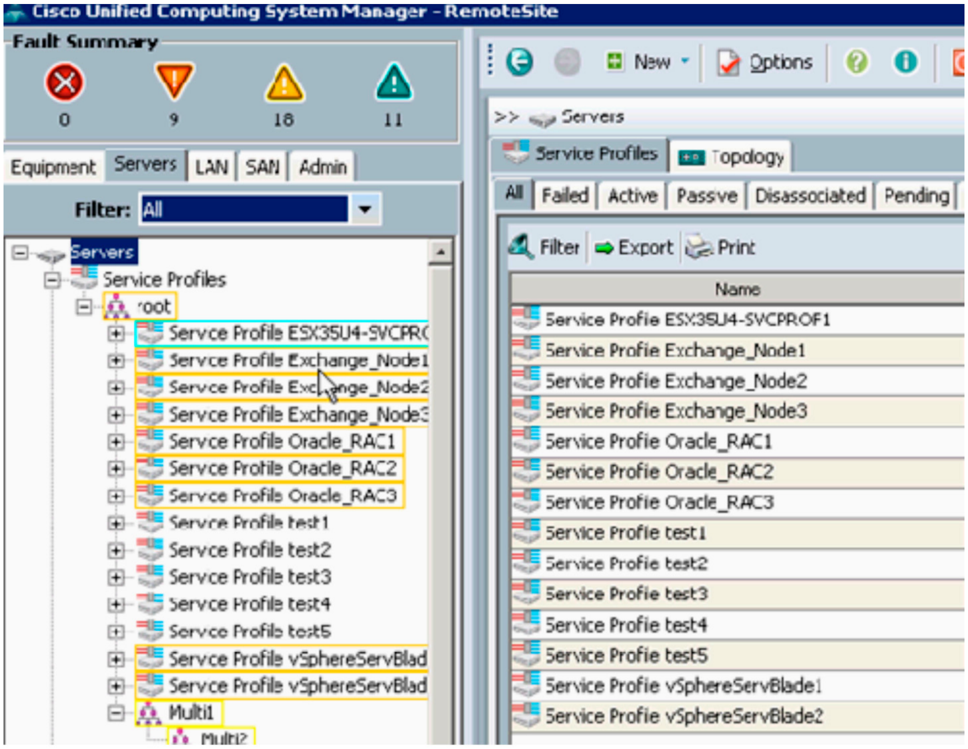
User: root

Password:

The merge option should be selected since the goal is to retain the existing configuration of the D2 domain and just add the service profiles and associated policies from Cisco Unified Computing System D1 to the D2 configuration.

Figure 44 shows the results of the merge operation. All the service profiles from Cisco Unified Computing System D1 are now present in the Cisco UCS Manager hierarchy on Cisco Unified Computing System D2.

Figure 44. Cisco UCS Manager Display of Service Profiles

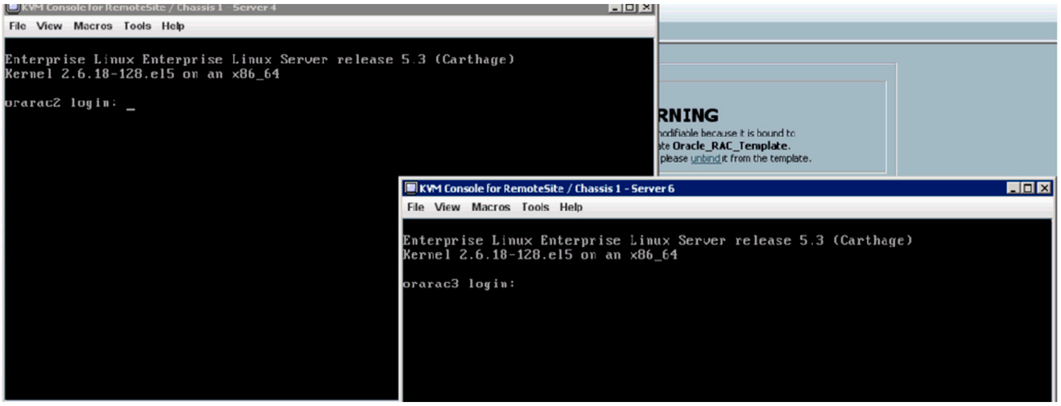


Name
Service Profile ESX35U4-SVCPROF1
Service Profile Exchange_Node1
Service Profile Exchange_Node2
Service Profile Exchange_Node3
Service Profile Oracle_RAC1
Service Profile Oracle_RAC2
Service Profile Oracle_RAC3
Service Profile test1
Service Profile test2
Service Profile test3
Service Profile test4
Service Profile test5
Service Profile vSphereServBlade1
Service Profile vSphereServBlade2

The service profiles will automatically associate with available blades as long as they are in the same slot (chassis slot number) as they were in the primary domain and are not already associated with another service profile.

Figure 45 shows the console from two of the RAC nodes after they are automatically rebooted as part of the service profile association process.

Figure 45. KVM Console Showing That Oracle Is Ready



KVM Console for RemoteSite / Chassis 1 - Server 4
File View Macros Tools Help

```
Enterprise Linux Enterprise Linux Server release 5.3 (Carthage)
Kernel 2.6.18-128.el5 on an x86_64

orarac2 login: _
```

RNING
nodifiable because it is bound to
ate Oracle_RAC_Template.
please unbind it from the template.
KVM Console for RemoteSite / Chassis 1 - Server 6
File View Macros Tools Help

```
Enterprise Linux Enterprise Linux Server release 5.3 (Carthage)
Kernel 2.6.18-128.el5 on an x86_64

orarac3 login:
```

The Oracle RAC instance will not automatically start since the devices associated with Oracle CRS are different at the disaster recovery site. To get the cluster services to start, you need a simple script that renames the PowerPath device names to match those in the primary domain. This script is included in Appendix B in this document.

Note that the situation described here is a limitation of the Oracle RAC services framework itself. Cluster services, such as Microsoft clusters, that depend on the signatures on the devices and not the device name to determine the cluster configuration do not need any additional intervention at the disaster recovery site.

As a best practice, the block of KVM IP addresses that were imported from the primary Cisco Unified Computing System domain should be removed as those addresses may not be valid at the disaster recovery site.

Conclusion

The new architecture that the Cisco Unified Computing System brings to the market does not hinder the operation of products such as EMC PowerPath and TimeFinder software, but rather integrates with them, bringing tremendous additional value to IT organizations. Used in combination with EMC's synchronous data protection and cloning technology and the best practices of both companies' technology stacks, the Cisco Unified Computing System provides a new disaster recovery paradigm that augments current industry practices, increasing their value.

TimeFinder used together with Cisco Unified Computing System service profile templates enables rapid provisioning of new compute nodes on an existing Oracle RAC instance. This capability allows organizations to add additional compute power when necessary instead of having to overengineer from the start and incur additional costs.

Together, EMC SRDF/S, the Cisco Unified Computing System import and export and open configuration (XML) capabilities, and the new EMC Symmetrix VMAX enable innovative approaches to moving applications from site to site in the event of a disaster. This combined solution streamlines operations and allows much more efficient use of DR assets on a continual basis.

Appendix A: Procedure to Add a Node to an Existing Oracle Cluster

The following steps show the process for adding an Oracle node to an existing Oracle RAC instance on the Cisco Unified Computing System. The starting point is the successful clone of the binaries for Oracle CRS and the Oracle database. The cluster in this example consists of **orarac1** and **orarac2**, with the host **orarac3** used as the third node. The oracle user is **oracle** with a group of **oinstall**. The commands need to be run as the user and on the host designated. For those commands requiring an X-terminal display, a virtual networking computing (VNC) server is used. Steps 1 through 9 are for cloning CRS, and Steps 10 through 14 are for adding Oracle ASM and the database.

1. As **root** on **orarac3**, run the following code to clean up the directories:

   ```
   cd/orabin/app/oracle/product/11.1.0/crs
   rm –rf ./opt/oracle/product/11g/crs/log/orarac*
   find . -name '*.ouibak' -exec rm {} \;
   find . -name '*.ouibak.1' -exec rm {} \;
   rm –rf root.sh*
   cd cfgtoollogs
   find . -type f -exec rm -f {} \;
   chown –R oracle:oinstall/orabin/app/oracle/product/11.1.0/crs
   rm -rf/orabin/app/oraInventory
   mkdir -p/orabin/app/oraInventory
   chown oracle:oinstall/orabin/app/oraInventory
   rm -rf/etc/oracle/scls_scr
   rm -f/etc/inittab.crs
   rm -rf/var/tmp/.oracle
   rm -rf/tmp/.oracle
   ```

2. As **root** on **orarac3**, run the following code to preupdate:

   ```
   ./orabin/app/oracle/product/11.1.0/crs/install/preupdate.sh -crshome /orabin/app/oracle/product/11.1.0/crs -crsuser
   oracle -noshutdown
   ```

3. As **oracle** (assuming that environment variables such as ORA_CRS_HOME are set) on **orarac3**, run the following code to clone the inventory:

   ```
   perl/orabin/app/oracle/product/11.1.0/crs/clone/bin/clone.pl
   ORACLE_BASE=/orabin/app/oracle ORACLE_HOME=$ORA_CRS_HOME
   ORACLE_HOME_NAME=11gCrs "sl_tableList={orarac3:orarac3-priv:orarac3-vip}"
   INVENTORY_LOCATION=/orabin/app/oraInventory -noConfig
   ```

4. As **root** on **orarac3** (if the file is not present, copy it from another node, though this script is not critical - it just changes some permissions that should already be changed), run this code:

   ```
   ./orabin/app/oraInventory/orainstRoot.sh
   ```

5. As **oracle** on **orarac1**, run this code to add the new cluster node:

   ```
   $ORA_CRS_HOME/oui/bin/addNode.sh –silent "CLUSTER_NEW_NODES={orarac3}"
   "CLUSTER_NEW_PRIVATE_NODE_NAMES={orarac3-priv}"
   "CLUSTER_NEW_VIRTUAL_HOSTNAMES={orarac3-vip}" –noCopy
   ```

6. As **root** on **orarac1**, run this code to update the inventory:

   ```
   /orabin/app/oracle/product/11.1.0/crs/install/rootaddnode.sh
   ```

7. As **root** on **orarac3**, run this code to change permissions (use the defaults; do not overwrite anything):

/orabin/app/oracle/product/11.1.0/crs/root.sh

8. As **oracle** on **orarac1**, run this code to configure Oracle Notification Service (ONS):

   onsconfig add_config orarac3:remote_port

   The remote port must be located by looking on **orarac3** in the $ORA_CRS_HOME/cfgtoollogs/configToolAllCommands file.

9. As **oracle** on **orarac3**, run this code to verify cloning:

   /orabin/app/oracle/product/11.1.0/crs/bin/cluvfy stage -post crsinst -n orarac3

10. As **oracle** on **orarac3**, run this code to clone ORACLE_HOME:

    perl/orabin/app/oracle/product/11.1.0/db_1/clone/bin/clone.pl '-

    "CLUSTER_NODES={orarac1,orarac3}"' '-O"LOCAL_NODE=orarac3"'

    ORACLE_BASE=/orabin/app/oracle ORACLE_HOME=$ORACLE_HOME

    ORACLE_HOME_NAME=11gOracle '-O-noConfig'

11. As **oracle** on **orarac1**, run this code:

    ./$ORACLE_HOME/oui/bin/runInstaller –updateNodeList ORACLE_HOME=$ORACLE_HOME "CLUSTER_NODES={orarac1,orarac2,orarac3}"

12. As **root** on **orarac3**, run this code:

    ./orabin/app/oracle/product/11.1.0/db_1/root.sh

13. As **oracle** on **orarac3**, run the following code in a VNC session (requires a GUI):

    ./orabin/app/oracle/product/11.1.0/db_1/bin/netca

    Choose **orarac3** and accept all the defaults in the configuration. You will only be allowed to add a listener.

14. As **oracle** on **orarac3**, run the following code in a VNC session (requires a GUI):

    ./orabin/app/oracle/product/11.1.0/db_1/bin/dbca

    Choose ASM Management. You will be prompted to create a new instance on the new node. Accept all the defaults.

    After this process is complete, select Instance Management as the next option and choose **orarac3** from the list. Accept all the defaults to create the new instance.

Appendix B: PowerPath Device Naming Script

The script “Current_mapping.sh” is executed on the primary site to create the mappings file called emcpowerdev_mappings. This mapping files is then used by the restore portion of the script to rename the PowerPath device names on the secondary EMC system. This is necessary due to device naming dependencies for some of the Oracle Cluster Ready Services processes.

```
#!/bin/bash
#
#   The script requires SE installed and licensed on the server
#
#  First find the mapping on RAC server
#
powermt display dev=all | egrep 'Pseudo|device ID' | cut -f2 -d"=" \
    | xargs -n 2 > emcpowerdev_mappings
#
if (( $? == 0 )); then      # determining current mapping succeeded
  cat emcpowerdev_mappings | awk '{print $1 " " $2}' | while read power dev
  do
    #
    #  Determine the R2 device. The device name will be used to rebuild on DR
    #
    ans=$(symdev -sid 1351 show $dev | grep "Remote Device Symmetrix Name" \
                | awk -F ':' '{print $2}')
    #
    #   Print the powerdevice R1 and R2 and store it in a temporary file
    #
    if (( $? == 0 )); then
      echo $power $dev $ans >>/tmp/emcpower.$$
    else
      echo "Determining the R2 device failed. Exiting..."
      exit
    fi
  done
```

```
else
    echo "Power device to R1 device failed. Exiting..."
    exit
fi
#
#  save the original file and move the temporary file
#
if (( $? == 0 )); then
  if [[ -f emcpowerdev_mappings ]]; then
    /bin/mv emcpowerdev_mappings emcpowerdev_mappings.bkp
  fi
  #
  if (( $? == 0 )); then
    /bin/mv/tmp/emcpower.$$ emcpowerdev_mappings
  else
    echo "Renaming of the original file failed...Exiting..."
  fi
else
  echo "Could not create the mapping file...Exiting..."
fi
```

Restore Mapping.sh

Restore_mapping.sh is executed on the target site to recreate the mappings. The script uses the emcpowerdev_mappings file that was created at the primary site.

```
#!/bin/bash
#
#   The script requires SE installed and licensed on the server
#
#  Prompt user for a free emcpower device to use when shuffling names around
#
#   This can be determined by running the command:
#
```

```
#   emcpadm getfreepseudo
#
#   and select one of the free device from the list that is returned
#
echo -n "Provide free emcpower psuedo device name: "
read free
#
#  First find the mapping on RAC server
#
powermt display dev=all | egrep 'Pseudo|device ID' | cut -f2 -d"=" \
    | xargs -n 2 > current_mapping
#
#  release all PowerDevices that are in use but not available
#
powermt release
#
if (( $? == 0 )); then
   cat current_mapping | awk '{print $1 " " $2}' | while read power dev
   do
     #
     #  Determine the correct mapping
     #
     enew=$(grep ${dev}$ emcpowerdev_mappings | awk '{print $1}')
     #
     #   Rename the original powerdevice to a free name. This frees up the name
     #   name for the correct device to be named to the origianl powerdevice.
     #
     echo "Device: $dev"
     echo "Current Name: $power"
     echo "Original Name: $enew"
     #
     emcpadm renamepseudo -s $enew -t $free >/dev/null 2>&1
     #
     #  Rename the current powername to the original/correct powername
     #
     emcpadm renamepseudo -s $power -t $enew >/dev/null 2>&1
     #
     # Finally rename the device that was moved to free name back to the
     # newly freed name.
     #
     emcpadm renamepseudo -s $free -s $power >/dev/null 2>&1
   done
fi
```

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 1-408-526-4000 or 1-800-553-NETS (6387)
Fax: 1-408-527-0883

EMC Corporation
176 South Street
Hopkinton, MA 01748
USA
www.emc.com
Tel: 508-435-1000

(1002R) C11-599794-00 05/10