Oracle 9i RAC Deployment on AIX Using VERITAS Storage Foundation for Oracle RAC 4.0 as Clusterware

R. Ballough
ballough@us.ibm.com
IBM Advanced Technical Support
September 6, 2004
Introduction......................................................................................................................... 4
Storage Foundation for Oracle RAC 4.0 Requirements .......................................................... 5
Storage Foundation for Oracle RAC 4.0 Architecture Overview......................................... 7
  Cluster Communication/Group membership .............................................................. 7
  I/O Components .......................................................................................................... 9
  Shared Disk Subsystems ............................................................................................. 9
  I/O Fencing ............................................................................................................... 10
  Dynamic Multi-Pathing (DMP) ............................................................................... 11
  VERITAS Volume Manager (VxVM) ..................................................................... 11
  Cluster Volume Manager (CVM) ......................................................................... 12
  CFS ........................................................................................................................... 12
  Oracle Disk Manager ............................................................................................... 12
  VCS Service Groups ............................................................................................... 14
Quick Installation Guide................................................................................................... 15
Server Preparation........................................................................................................... 16
  Operating System Setup ......................................................................................... 16
  Tunable parameter changes ..................................................................................... 16
  Environment Variables ............................................................................................. 17
Obtain license key for SFRAC ....................................................................................... 17
Create Oracle user and DBA group .............................................................................. 17
Set up network communication ..................................................................................... 18
Configure Network parameters ..................................................................................... 18
Allocate Storage ............................................................................................................ 19
Install and Configure SFRAC software ......................................................................... 19
  Verify installation ..................................................................................................... 21
  Configure I/O fencing ............................................................................................... 21
  Add the “UseFence” cluster attribute to the main.cf file ........................................ 22
Create disk groups, volumes, and filesystems .............................................................. 23
  Oracle Binaries ........................................................................................................ 23
  Data Volumes & Filesystems ............................................................................... 24
Run pre-install tasks ...................................................................................................... 25
Install 9i ........................................................................................................................ 25
Apply current patch set .................................................................................................. 33
Run post-install tasks .................................................................................................... 36
Configure service groups .............................................................................................. 37
Set up demo database .................................................................................................... 39
Configure Listeners................................................................................................... 39
Create demo database ............................................................................................... 42
Transparent Application Failover (TAF) demo .......................................................... 48
  Install Oracle 9i Client to Windows client machine ................................................. 49
  Install TAF Sample Code......................................................................................... 52
Appendix A: Differences between Storage Foundation for Oracle RAC 4.0 for AIX vs.
  Solaris ............................................................................................................................... 56
Reminders ......................................................................................................................... 57
Trademarks ......................................................................................................................... 58
References......................................................................................................................... 59
Introduction

VERITAS Storage Foundation for Oracle Real Application Clusters (SFRAC) provides an enabling clusterware on which to run Oracle RAC on AIX, as an alternative to HACMP or GPFS.

This document is written in two parts; the first section covers the SFRAC requirements and provides an architectural overview of the VERITAS SFRAC product. The second section is a quick install guide outlining the installation and setup of a two-node RAC cluster on AIX.

The audience is assumed to be familiar with the basic components of clustering products, have a basic understanding of Oracle RAC, and be familiar with SAN-attached storage.

This document is intended to supplement, not replace, configuration guides which should be used as references for an Oracle 9i RAC installation on AIX, such as the materials listed below:

http://support.veritas.com
- VERITAS Storage Foundation™ 4.0 for Oracle RAC – Release Notes (AIX)
- VERITAS Storage Foundation™ 4.0 for Oracle RAC – Installation and Configuration Guide (AIX)

http://docs.oracle.com
- Oracle 9i Installation Guide Release 2 for UNIX Systems
- Oracle 9i Release Notes Release 2 (9.2.0.1.0) for AIX –Based 5L systems
- Oracle 9i Real Application Clusters Guide for UNIX Systems
Storage Foundation for Oracle RAC 4.0 Requirements

The latest information about supported hardware and software can be found in the VERITAS document “Hardware Compatibility List (HCL) for Storage Foundation and High Availability Products on AIX”, which is located at http://support.veritas.com.

Current requirements for filesets and APARs are located in the Storage Foundation Release Notes, also available at http://support.veritas.com.

At the time of this document, the following are requirements for installation of Storage Foundation for Oracle RAC:

- 64-bit pSeries or RS/6000 hardware
- Server systems with a minimum of 1 GB RAM. Two or more CPUs are recommended, plus 7 GB disk space for VERITAS and Oracle installation.
- 2-4 cluster nodes
- A minimum of two Ethernet connections are required for the RAC/cluster interconnect; three are recommended.
- SAN-attached storage must support SCSI-3 Persistent Reservations. The following storage subsystems are currently supported:
  - ESS F20, firmware level 2.3.2.70 or above
  - ESS 800, firmware level 2.3.2.71 or above
  - EMC Symmetrix DMX Series
  - EMC Symmetrix 8000 Series
  - Hitachi 9900 series, firmware 21-06-22-00/00 or above. Additional setup steps are required for Hitachi, see “Veritas Storage Foundation 4.0 for Oracle RAC Release Notes” for details
- No third-party multipathing drivers are currently supported; multipathing functionality must be provided by VERITAS Dynamic Multipathing (DMP)
- Three separate coordinator LUNs are required to support I/O fencing operations. Each of these LUNs can be as small as 1 MB in size.
- AIX 5.2 ML 3 64-bit kernel only
- APAR IY56497
- Fileset xlC.aix50.rte (6.0.0.7 or above)
- Additionally, Oracle requires the following filesets at the levels listed, or above:
  - Bos.adt.libm 5.2.0.10
  - Bos.perf.tools 5.2.0.10
  - Perfagent.tools 5.2.0.10
  - Rsct.core 2.3.1.1
  - Rsct.compat.basic 2.3.1.1
  - Rsct.compat.clients 2.3.1.1
  - Rsct.basic 2.3.1.1
- Oracle 9i version 9.2.0.5 or above, plus patch # 3734279
Coexistence with other software products:
VERITAS SFRAC cannot coexist with HACMP. Single-instance Oracle databases using LVM may coexist with Oracle instances using VERITAS products.

A typical 2-node configuration example is shown below.
Storage Foundation for Oracle RAC 4.0 Architecture Overview

The VERITAS Storage Foundation 4.0 for Oracle RAC is a set of products which enables Oracle RAC to run in a clustered environment using the VERITAS product suite, rather than HACMP and/or GPFS. VERITAS Storage Foundation 4.0 for Oracle RAC is built on VERITAS Cluster Server (VCS), VERITAS Cluster Volume Manager (CVM), VERITAS Filesystem (CFS), and VERITAS Storage Foundation, which includes the base VERITAS Volume Manager (VxVM) and filesystem (VxFS).

Cluster Communication/Group membership

The VERITAS Cluster Server (VCS) provides two components for heartbeat, cluster communication messaging, and cluster group membership information, which are called the Low Latency Transport (LLT) and the Group Atomic Broadcast (GAB).

LLT is a proprietary transport/network layer (over Ethernet network interfaces as the data link layer) used in place of TCP or UDP over IP. LLT provides a reliable transport mechanism which is also low latency, as its packet configuration is optimized for cluster traffic. LLT load balances cluster communications for performance and fault resilience over all Ethernet connections provided (three are recommended, and up to eight can be used). LLT is also responsible for sending out heartbeat communications.
Oracle 9i RAC uses inter-process communication protocols (IPC) to transfer data blocks and messaging between instances. SFRAC provides a shared library, VCSIPC, to handle efficient messaging of RAC IPC traffic over multiple physical network connections using a multiplexer for LLT called LMX.

The Group Atomic Broadcast (GAB) is responsible for monitoring heartbeat communication between the cluster nodes, and updating cluster membership information accordingly. GAB also distributes information to all cluster nodes, for example, volume management information from CVM or filesystem information for CFS.

Since the Oracle Cluster Manager process, ORACM, also needs to know the membership state of the nodes, GAB provides this information to ORACM via a Membership Manager, VCSMM, which interfaces with GAB.

The following chart depicts the processes which utilize GAB and LLT:
**I/O Components**

The following diagram shows the data stack for 9i RAC on VERITAS Storage Foundation for Oracle RAC:

Oracle 9i RAC processes read and write data through the Oracle Disk Manager (ODM), which interfaces with the VERITAS Cluster File System (CFS). CFS then accesses the physical storage layer through with the Cluster Volume Manager (CVM), taking advantage of DMP for multipathing.

**Shared Disk Subsystems**

Oracle 9i RAC requires simultaneous access to database storage from all cluster members, thus at the lowest level a shared disk subsystem is required. Since I/O fencing mechanisms are provided using SCSI-3 persistent reservations, all disk storage systems which are certified or will be certified must support SCSI-3 PR. See the
“Storage Foundation for Oracle RAC 4.0 Requirements” section for information on appropriate disk choices.

I/O Fencing

In a cluster configuration, I/O fencing is used to prevent data access by a failed cluster node. VERITAS SFRAC implements I/O fencing on AIX using SCSI-3 Persistent Reservation; a method of determining a node’s right to access disk based on the presence of a registration “key”. Once nodes register a key with the SCSI-3 device, they can establish a reservation to the device. If the registration key is removed, the node can no longer access the device.

With VERITAS SFRAC, VxVM assigns the registration keys, and a driver, vxfen, keeps track of which nodes have reservation access to which disks. In the event of a split-brain condition in the cluster, the individual nodes race for control for the three coordinator LUNs. The node or nodes which gain control then fence out inaccessible nodes by removing their reservation keys. In the case where more than two nodes form a cluster, optimization exists in the vxfen driver to favor the largest subcluster to remain.

The following table from the VERITAS SFRAC Install & Configuration Guide shows the various possible failure scenarios in a two-node cluster, and how I/O fencing works in these scenarios.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Both private networks fail</td>
<td>Node A races for majority of coordinator disks. If Node A wins race for coordinator disks. Node A ejects Node B from the shared disks and continues</td>
<td>Node B races for majority of coordinator disks. If Node B loses the race for the coordinator disks, Node B removes itself from the cluster.</td>
</tr>
<tr>
<td>Both private networks function again after event above.</td>
<td>Node A continues to work.</td>
<td>Node B has crashed. It cannot start the database since it is unable to write to the data disks.</td>
</tr>
<tr>
<td>One private network fails.</td>
<td>Node A prints message about an IOFENCE on the console but continues.</td>
<td>Node B prints message about an IOFENCE on the console but continues.</td>
</tr>
<tr>
<td>Node A hangs.</td>
<td>When Node A is no longer hung, any queued writes to the data disks fail because Node A is ejected. When Node A receives message from GAB about being ejected, it removes itself from the cluster.</td>
<td>Node B loses heartbeats with Node A, and races for a majority of coordinator disks. Node B wins race for coordinator disks and ejects Node A from shared data disks.</td>
</tr>
<tr>
<td>Nodes A and B and private networks lose power. Coordinator and data disks retain power. Power returns to nodes and they restart, but private networks still have no power.</td>
<td>Node A restarts and I/O fencing driver (vxfen) detects Node B is registered with coordinator disks. The driver does not see Node B listed as member of cluster because private networks are down. This causes the I/O fencing device driver to prevent Node A from joining the cluster.</td>
<td>Node B restarts and I/O fencing driver (vxfen) detects Node A is registered with coordinator disks. The driver does not see Node A listed as member of cluster because private networks are down. This causes the I/O fencing device driver to prevent Node B from joining the cluster.</td>
</tr>
<tr>
<td>NodeA crashes while Node B is down. Node B comes up and Node A is still down.</td>
<td>Node A is crashed.</td>
<td>Node B restarts and detects Node A is registered with the coordinator disks. The driver does not see Node A listed as member of cluster.</td>
</tr>
</tbody>
</table>
The disk array containing two of the three coordinator disks is powered off.

Node A continues to operate as long as no nodes leave the cluster.

Node B continues to operate as long as no nodes leave the cluster.

Node B leaves the cluster and the disk array is still powered off.

Node A races for a majority of coordinator disks. Node A fails because only one of three coordinator disks is available. Node A removes itself from the cluster.

Node B leaves the cluster.

Dynamic Multi-Pathing (DMP)

Dynamic Multi-Pathing (DMP) is part of VERITAS’ Storage Foundation product, and offers support both for path failover and load balancing between multiple fibre channel connections to disk.

VERITAS Volume Manager (VxVM)

The VERITAS Volume Manager organizes groups of LUNs into disk groups. Within individual disk groups, disk space is organized into logical objects called volumes, plexes, and subdisks. A subdisk is a disk segment, which can be an entire LUN or a piece of a LUN. A plex is an ordered collection of one or more subdisks; when more than one subdisk is used, they can be used to form a plex which is either concatenated or striped. A volume is a collection of plexes that appears to the system as a virtual device (block and character), which can then be accessed as a raw device, or can be a device on which to place a filesystem. If a volume has more than one plex, it is mirrored.

In the example below, PLEX 1 is comprised of two subdisks on two separate LUNs, which could be either striped or concatenated. PLEX 2 is comprised of two subdisks on the same physical LUN, so only a concatenated configuration would be advisable. If a volume was created out of PLEX 1 and PLEX 2, it would be a mirrored volume. If a volume was created out of PLEX 1, it would not be mirrored, since there is only one copy of the underlying data.

The configuration of logical volumes is maintained by the volume manager configuration daemon, vxconfigd.
Cluster Volume Manager (CVM)

CVM is essentially a multi-node extension of VxVM, allowing the sharing of data volumes among multiple cluster nodes. Each cluster node runs its own vxconfigd daemon, and the first node to enter the cluster will own the master vxconfigd process. All other cluster nodes will run slave vxconfig processes. The master can be determined by using the command “vxdctl –c mode”.

When a change is made to a volume, the master vxconfigd process pushes out the changes (using GAB and LLT) out to the slave vxconfigd processes on other nodes. The configuration master must be used for updating shared volume configuration data.

CVM does not impose any type of write locking; data integrity is expected to be handled by Oracle RAC.

CFS

Similarly to CVM, CFS is also an extension of its single-node counterpart, the VERITAS File System (VxFS), which allows simultaneously mounting filesystems on multiple cluster nodes. As with CVM, CFS also operates in a master/slave architecture, where the master node carries out any change operation to the filesystem (creating or deleting, for example), then propagates the changes to the slave nodes. Data access, however, happens directly from any node which has the filesystem mounted.

When Oracle RAC is not used, CFS uses a distributed locking mechanism, called the Global Lock Manager (GLM), to handle write locking of data, as well as consistency in filesystem metadata and cache coherency of the filesystem buffer cache. Like vxconfigd, GLM pushes its changes out over the LLT interface.

However, with the use of Oracle RAC, the GLM only handles filesystem metadata consistency – the Oracle Disk Manager is used instead, which bypasses the file system buffer cache, and takes over the role of handling data locking.

With an Oracle RAC configuration, any type of Oracle file, including SORACLE_HOME, can reside on CFS filesystems.

Oracle Disk Manager

As part of Oracle 9i, Oracle has developed a new disk and file management API called Oracle Disk Manager (ODM). ODM allows the Oracle to allocate and release disk space, and read or write disk blocks directly. ODM is advertised to improve file I/O in three major areas: filesystem performance on par with raw, reduced system overhead, and simplified file administration.
With ODM, at the time the database is opened, all files are opened using the odm_identify() call, which assigns an identifier to each file and stores this information in the SGA. Subsequently, the ODM issues all I/O requests, both reads and writes, using the same odm_io() call, regardless of the underlying data file type (raw or filesystem). An async option for use of kernelized async IO (kaio) is also available. The odm_io() call replaces the use of a variety of calls to handle distinct i/o types, such as readv(), pread64(), pwrite64(), lio_listio(), aio_read(), etc. Odm_io() requests can also be batched for multiple read/writes, such as for writing out dirty buffers, rather than executing multiple i/o calls as is required without ODM.

There appear to be many potential performance advantages to the ODM implementation. As fewer I/O calls are issued, system resources are saved. Using the same I/O call for filesystem and raw I/O both bypasses the filesystem write lock, and results in raw I/O performance from block devices. And, as the data files are opened at database open and centrally stored in the SGA, per process file descriptors are no longer used, and Oracle processes no longer have the overhead of opening and closing files, resulting in lower system overhead, as well as removing limits on the number of open files. One particular benchmark of an Oracle OLTP Inventory Management Workload showed an 8% reduction in kernel mode CPU usage – over raw, non-ODM data storage.

ODM also offers advantages in the area of file creation and management. ODM allocates space in a contiguous fashion, using the odm_create() command, as well as supporting the use of Oracle’s autoextend by allowing the addition of contiguous disk blocks to tablespaces via the odm_resize() command. When the space allocation is successful, the space is committed using odm_commit(), or in the event the allocation fails, ODM performs the necessary cleanup using the odm_abort() command.

A third party vendor’s ODM driver, such as VRTSodm, is needed to implement ODM, and Oracle 9i will automatically use the ODM API when the ODM driver, installed as part of the VERITAS install process, is present. The following message will appear in the Oracle alert log at database startup: “Oracle instance running with ODM”

A short note on VERITAS Quick I/O: Often in the discussion of raw devices versus filesystems, experienced VERITAS users will inquire about the presence of Quick I/O (QIO). QIO performs similar functionality as the use of ODM files, providing unbuffered asynchronous I/O; however, it requires additional setup and maintenance which is not needed with ODM, such as special file creation for all QIO files, and special handling for backups. For this reason, and due to integration with Oracle’s APIs, Oracle considers ODM the preferred solution over QIO when used in an Oracle 9i environment on a supported platform.
VCS Service Groups

Although application failover is typically not configured in an Oracle RAC environment, the VERITAS Cluster Server can be configured to control startup and shutdown of the components needed for RAC. Resource definitions which are logically grouped together for purposes of startup and shutdown are called service groups. The service group configuration file is /etc/VRTSvcs/conf/config/main.cf, which is a text file that can be edited as desired.

Startup and shutdown agents for CVM, CFS, and Oracle are provided with Storage Foundation for Oracle RAC.
Quick Installation Guide

The following sample installation is for the two-node cluster configuration depicted in the “requirements” section of this document. The two nodenames in this example will be “erpcc2” and “erpcc8”, and the cluster name will be rac_cluster. The primary installation tasks are outlined in the flow chart below:

- Server Preparation
  - Configure Network
    - Install SFRAC Software
      - Verify Installation
        - Configure I/O Fencing
          - Create CVM Disk Groups, Volumes, and CFS Filesystems
            - Install Oracle 9i RAC
              - Perform Post Installation Tasks
                - Create Database
Server Preparation

Operating System Setup

Since VERITAS software does not encapsulate the rootvg, typical operating system preparation tasks involving the rootvg must be done using AIX commands.

All tasks in this section must be performed on both cluster nodes.

Ensure sufficient paging space is available

- root# lsps –a


Mirror rootvg (assuming hdisk0 is the current boot disk, and hdisk1 is available):

- root# mirrorvg rootvg hdisk1
- root# bosboot –ad /dev/hdisk1
- root# bootlist –m normal hdisk0 hdisk1

Check for patches and efixes listed in the “requirements” section, install if needed:

- root# instfix –ik <patch #>

Patches and maintenance levels can be downloaded from [http://techsupport.services.ibm.com](http://techsupport.services.ibm.com), and installed using the “smit install” menu

Check the filesets levels listed in the “requirements” section:

- root# lslpp –L

Change to a 64-bit kernel

- Check to see if the kernel is 32 or 64-bit:
  
  root# bootinfo –K

  - If the kernel is 32-bit, change to 64-bit:
    
    root# ln –sf /usr/lib/boot/unix_64 /unix
    root# ln –sf /usr/lib/boot/unix_64 /usr/lib/boot/unix
    root# bosboot –ad /dev/hdisk0 # assuming hdisk0 is the boot device
    root# shutdown –Fr

Tunable parameter changes

AIOservers (use ‘smit aio’ to change):

As ODM uses kernelized async I/O (aio), aioservers need to be enabled, but the minserver and maxserver values do not need to be increased from the defaults:
• Minserver=(default value)
• Maxserver=(default value)
• Maxreqs=4*#disks*queue_depth  (do not decrease below 4096)
• Enable at system restart.

Monitor periodically using `pstat -a | grep aios | wc -l`

Virtual Memory Manager (VMM) initial settings:

• Minperm%  5%
  o root# vmo -p -o minperm%=5
• Maxperm%  20%
  o root# vmo -p -o maxperm%=20
• Maxclient% Set equal to maxperm
  o root# vmo -p -o maxclient%=20

For a complete discussion on VMM and other tuning parameters, refer to the IBM Redbook “Database Performance Tuning on AIX”, available for download from http://www.redbooks.ibm.com.

Environment Variables

Verify that the following directories are included in the PATH environment variable:

• PATH=/usr/sbin:/sbin:/usr/bin:/usr/lib/vxvm/bin:/opt/VRTSvxfs/sbin:/opt/VRTSvcs/bin:/opt/VRTS/bin:/opt/VRTSvcs/rac/bin:/opt/VRTSob/bin:$PATH; export PATH

Include VERITAS man pages:

• MANPATH=$MANPATH:/opt/VRTS/man; export MANPATH

**Obtain license key for SFRAC**

Access the web site http://vlicense.veritas.com using the information provided in the License Key Request Form, to generate a license key for SFRAC.

**Create Oracle user and DBA group**
Create dba group on each cluster node:
• root# mkgroup –A id=1000 adms=’root’ dba

Create Oracle user on each cluster node:
• root# mkuser id=1000 pgrp=’dba’ groups=dba home=/sfrac/oracle’ oracle
• root# mkdir /sfrac/oracle
• root# chown oracle:dba /sfrac/oracle
Allocate a password to Oracle user:
• root# passwd oracle

Edit /etc/security/limits and set limits for Oracle user to unlimited (-1)

Set up network communication

On each node, configure en0 as the public Ethernet interface, and configure en1, en2, and en3 as the cluster interfaces.
Edit the following files on each node:

• /etc/hosts
  <erpcc2 en0 IP address>     erpcc2
  <erpcc8 en0 IP address>     erpcc8

• /etc/hosts.equiv
  erpcc2     root
  erpcc8     root
  erpcc2     oracle
  erpcc8     oracle

• .rhosts and /home/oracle/.rhosts
  “*”
  Note: it is possible to use ssh, rather than .rhosts, but as the setup is more complex, use of .rhosts is recommended

Verify the ability to rcp a file between systems for root and Oracle

Configure Network parameters

• Set sb_max >= 1 MB (1048576)
• Set tcp_sendspace = 262144
• Set tcp_recvspace = 262144
• Set rfc1323 = 1 (enable)
• Set maxmbuf (on sys0) = 0

Typically with HACMP + RAC, udp_sendspace and udp_recvspace parameters should be increased; however, SFRAC doesn’t use the UDP protocol, and iptrace results confirm that no UDP packets are sent on the cluster interconnects, so these parameters only need to be set if needed for non-cluster UDP traffic.

**Allocate Storage**

A minimum of three 1 MB LUNs must be configured for use as coordinator disks. These LUNs cannot be used for other purposes. Additionally, LUNs should be allocated for Oracle binaries and Oracle data. Refer to the disk vendor’s documentation for information regarding allocating a LUN to an AIX host.

**Install and Configure SFRAC software**

Mount the CD-ROM on the first cluster node:

- root# mkdir /cdrom
- root# mount -o ro -v cdrfs /dev/cd0 /cdrom

Run installation on the first cluster node only (for simplicity, only output which requires a response is shown below, with responses in **bold**):

- root# cd /cdrom/storage_foundation_for_oracle_rac
- root# ./installsfrac

Enter the system names separated by spaces on which to install SFRAC: **erpcc2 erpcc8**

Enter SFRAC License Key when prompted.

Are you ready to configure SFRAC? [y,n,q] (y)  **y**

Enter the unique cluster name: [?] **rac_cluster**

Enter the unique Cluster ID number between 0-255: [b,?]  **1**

Enter the NIC for the first private heartbeat NIC on erpcc2: [b,?]  **en1**

Would you like to configure a second private heartbeat link? [y,n,q,b,?] (y) **y**

Enter the NIC for the second private heartbeat link on erpcc2 [b,?]  **en2**

Would you like to configure a third private heartbeat link? [y,n,q,b,?] (y) **y**

Enter the NIC for the third private heartbeat link on erpcc2 [b,?]  **en3**

Would you like to configure a fourth private heartbeat link? [y,n,q,b,?] (y) **n**

Are you using the same NICs for private heartbeat links on all systems? [y,n,q,b,?] (y) **y**

Cluster Name: rac_cluster
Cluster ID Number: 1
Private Heartbeat NICs for :erpcc2 link1=en1 link2=en2 link3=en3
Private Heartbeat NICs for :erpcc8 link1=en1 link2=en2 link3=en3
Is this information correct? [y,n,q] (y)

The following information is required to add VCS users:
- A user name
- A password for the user
- User privileges (Administrator, Operator, or Guest)
Do you want to set the password for the Admin user (default password='password')? [y,n,q] (n)
Enter New Password:******
Enter Again:******
Do you want to add another user to the cluster? [y,n,q] (y)
User: admin Privilege: Administrators
Is this information correct? [y,n,q] (y)
Do you want to add another user to the cluster? [y,n,q] (y)
The following information is required to configure Cluster Manager:
- A public NIC used by each system in the cluster
- A Virtual IP address and netmask for Cluster Manager
Do you want to configure Cluster Manager (Web Console) [y,n,q] (y)
Active NIC devices discovered on erpcc2: en0
Enter the NIC for Cluster Manager (Web Console) to use on galaxy: [b,?] (en0)
Is en0 to be the public NIC used by all systems [y,n,q,b,?] (y)
Enter the Virtual IP address for Cluster Manager: [b, ?]

You can access the cluster manager using the user name “admin” and the password “password”
Cluster Manager (Web Console) verification:
- NIC: en0
- IP: <virtual ip address>
- Netmask: 255.255.240.0
Is this information correct? [y,n,q] (y)
Do you want to configure SMTP notification? [y,n,q] (y)

SFRAC filesets can be installed on systems consecutively or simultaneously. Installing filesets on systems consecutively takes more time but allows for better error handling.
Would you like to install Storage Foundation for Oracle RAC filesets on all systems simultaneously? [y,n,q,?] (n)
Do you want to start Storage Foundation for Oracle RAC processes now? [y,n,q] (y)

Full activation of Cluster Manager, Cluster Volume Manager, and Cluster File System requires a reboot on All Systems.
Execute /usr/sbin/shutdown -r to properly restart your systems

Do not reboot at this time

Do you want to set up the default disk group for each system? [y,n,?] (y)
Verify installation

At the end of the installation, the following message should be displayed:

Installation of Storage Foundation for Oracle RAC 4.0.0.0 has completed successfully.
The installation summary is saved at:
/opt/VRTS/install/logs/installsfrac223180217.summary
The installsfrac log is saved at:
/opt/VRTS/install/logs/installsfrac223180217.log
The installation response file is saved at:
/opt/VRTS/install/logs/installsfrac223180217.response

On each node, check for the presence of all of the required filesets:
• root# /opt/VRTSvcs/bin/chk_sfrac_pkgs

Check group membership information:

• root# gabconfig –a

Output should look similar to the following:

GAB Port Memberships
<table>
<thead>
<tr>
<th>Port</th>
<th>gen</th>
<th>a21901 membership 01</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>gen</td>
<td>a21904 membership 01</td>
</tr>
<tr>
<td>h</td>
<td>gen</td>
<td>a21903 membership 01</td>
</tr>
</tbody>
</table>

Configure I/O fencing

Run the command ‘vxdisk scandisks’, which scans all disk disk drives and updates VxVM and DMP. Note that this command must also be run any time additional LUNs are added to or deleted from the system:
• root# vxdisk scandisks

Identify the three 1 MB coordinator disks and use them to form the vxfencoordg disk group:
• root# vxdg init vxfencoorddg <disk1> <disk2> <disk3>

Test the coordinator disk group for persistent reserve:
• root# /opt/VRTSvcs/vxfen/bin/vxfentsthdw -c vxfencoorddg
• root# vxdg deport vxfencoorddg
• root# vxdg –t import vxfencoorddg
• root# vxdg deport vxfencoorddg
Create /etc/vxfendg on all cluster nodes:
  • root# echo “vxfencoorddg” > /etc/vxfendg

**Add the “UseFence” cluster attribute to the main.cf file**

Stop VCS
  • root# hastop –all

Backup configuration file main.cf:
  • root# cp /etc/VRTSvcs/conf/config/main.cf /etc/VRTSvcs/conf/config/main.orig

Edit /etc/VRTSvcs/conf/config/main.cf file and add “UseFence=SCSI3” line as in the example below:

```
cluster rac_cluster {
    UserNames = { admin = "cDRpdXPmHpzS." } 
    Administrators = { admin } 
    HacliUserLevel = COMMANDROOT 
    CounterInterval = 5 
    **UseFence = SCSI3**

) 
```

Verify the configuration of the file:
  • root# hacf -verify /etc/VRTSvcs/conf/config

Restart both cluster nodes using the ‘shutdown’ command:
  • root# shutdown –Fr

Check group membership information:
  
  • root# gabconfig –a

  Output should look similar to the following:

  **GAB Port Memberships**

  =====================================================
  Port a gen 4a1c0001 membership 01
  Port b gen g8ty0002 membership 01
  Port d gen 40100001 membership 01
  Port f gen f1990002 membership 01
  Port h gen d8850002 membership 01
  Port o gen f1100002 membership 01
  Port v gen 1fc60002 membership 01
  Port w gen 15ba0002 membership 01
gabconfig –a displays which cluster members have membership to the various components. “01” indicates node “0” and node “1”.

The following are the services which use the ports listed:

<table>
<thead>
<tr>
<th>Port</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>GAB</td>
</tr>
<tr>
<td>b</td>
<td>I/O fencing</td>
</tr>
<tr>
<td>d</td>
<td>ODM (Oracle Disk Manager)</td>
</tr>
<tr>
<td>f</td>
<td>CFS (Cluster File System)</td>
</tr>
<tr>
<td>h</td>
<td>VCS (VERITAS Cluster Server: high availability daemon)</td>
</tr>
<tr>
<td>o</td>
<td>VCSMM driver</td>
</tr>
<tr>
<td>v</td>
<td>CVM (Cluster Volume Manager)</td>
</tr>
<tr>
<td>w</td>
<td>vxconfigd (module for CVM)</td>
</tr>
</tbody>
</table>

**Create disk groups, volumes, and filesystems**

**Oracle Binaries**

The Oracle binaries can be either shared, that is, installed on a clustered filesystem, or they can be installed separately on each node. Examples in this paper will be shown with a shared binary configuration.

Find the master node:
- root# vxdctl –c mode
  mode: enabled: cluster active – MASTER

From the master node, create a disk group using the “-s” flag to designate a shared disk group:
- root# vxdg –s init oradg <disk>

Create a volume for the binaries:
- root# vxassist –g oradg make vol_orabin 7168M
- root# vxedit –g oradg set user=oracle group=dba vol_orabin

No additional volume is needed for SRVM data; this information can reside on this shared filesystem; however, if separate binaries are used, a shared volume (typically 300MB) must be configured to house this volume.

Make directory /sfrac/oracle on each cluster node:
- root# mkdir –p /sfrac/oracle

Run mkfs on the master VxVM node only:
- root# mkfs –V vxfs –o largefiles /dev/vx/rdsk/oradg/vol_orabin
Mount the filesystem from each cluster node:

- root# mount -V vxfs -o cluster -o largefiles /dev/vx/dsk/oradg/vol_orabin /sfrac/oracle

Change permissions from one cluster node:

- root# chown -R oracle:dba /sfrac/oracle
- root# chmod 755 /sfrac/oracle

**Data Volumes & Filesystems**

As there is a 1-to-many relationship between a disk group and one or more LUNs, use caution in creating disk groups such that entire LUNs are not wasted by creating too many disk groups.

For the sake of convenience (mine), I am creating one data volume to contain a demo database. For non-test configurations, planning should be done for a logical layout using an appropriate number of disk groups, volumes, and filesystems. Oracle’s Optimal Flexible Architecture section in the “Oracle 9i Administrator’s Reference for UNIX Systems” provides guidelines for filesystem standards.

- root# vxassist –g oradg make vol_oradata 7168M
- root# vxedit –g oradg set user=oracle group=dba vol_oradata

Create and mount the filesystems for the data volumes as follows:

Make directory /sfrac/oracle/oradata on each cluster node:

- root#mkdir –p /sfrac/oracle/oradata

Run mkfs on the master VxVM node only:

- root# mkfs –V vxfs –o largefiles /dev/vx/rdsk/oradg/vol_oradata

Mount the filesystem from each cluster node:

- root# mount –V vxfs -o cluster -o largefiles /dev/vx/dsk/oradg/vol_oradata /sfrac/oracle/oradata

Change permissions from one cluster node:

- root# chown –R oracle:dba /sfrac/oracle/oradata
- root# chmod 755 /sfrac/oracle/oradata

Tuning for VERITAS filesystems is performed using the ‘vxtunefs’ command, which can be examined in detail in either the vxtunefs man page, or in VERITAS reference manuals.
Run pre-install tasks

Create the srvConfig.loc file on all cluster nodes:

- root# mkdir /var/opt/oracle
- root# echo “srvconfig_loc=/sfrac/oracle/shared.dbf” > /var/opt/oracle/srvConfig.loc
- root# chown oracle:dba /var/opt/oracle/srvConfig.loc
- root# ln –s /etc/srvConfig.loc /var/opt/oracle/srvConfig.loc
- oracle$ touch /sfrac/oracle/shared.dbf

Run the VERITAS script preinstallora on the master VxVM node:

- root# cd /opt/VRTSvcs/rac/bin/scripts
- root# ./preinstallora

Install 9i

Make a staging area to hold the install disks:

- root# mkdir /orastage
- root# cd /orastage
- root# mkdir Disk1; mkdir Disk2; mkdir Disk3; mkdir Disk4
- root#chown –R oracle:dba /orastage

Copy Oracle disks to the created directories.

Log in as oracle, and set up the following environment variables in oracle’s .profile:

- export ORACLE_BASE=/sfrac/app/oracle
- export ORACLE_HOME=/sfrac/app/oracle
- export LIBPATH=$ORACLE_HOME/lib:/usr/lib:/lib
- export PATH=$PATH:$ORACLE_HOME/bin

Source the .profile file:

- oracle$ . ~/.profile

Set the DISPLAY environment variable to the appropriate value:

- oracle$ DISPLAY=host:0.0; export DISPLAY

Run the Oracle installer:

- oracle$ /orastage/Disk1/runInstaller
If this is the first time that the Oracle installer has been run on this system, a prompt will appear for an Inventory Location, where Oracle keeps information about installed products. The Oracle Inventory is typically installed in /oracle/oraInventory.
Since the Oracle binaries are shared, only one cluster node should be selected. For configurations using separate binaries, select both cluster nodes for installation.

Name the $ORACLE_HOME variable and enter a path to the desired directory to the Oracle binaries.
Select at least the Oracle Database and Real Applications Clusters.
Do not create the database as part of the install process.
The JDK location will probably be /usr/java131.
On each node, run root.sh as instructed:

- root# /sfrac/oracle/product/9.2.0$ ./root.sh
Cancel out of the Oracle Net configuration assistant.

Cancel out of EMCA.
The Oracle Enterprise Manager screen will launch automatically. Select File -> Exit.

**Apply current patch set**

NOTE: At the time of this document, 9.2.0.4 was the correct patch set, and is shown in the installation instructions below; however, the patch set for current deployments must be 9.2.0.5 or later.

Download oracle 9.2.0.4 to /orastage/9204

- oracle$ /orastage/9204/runInstaller

The first time OUI is launched, upgrade “Oracle Universal Installer 2.2.0.18”. After this is complete, close the installer, or the correct installer version will not be recognized. Relaunch the installer, then select “Oracle9iR2 Patch Set 9.2.0.4” for installation
As with the installation process, select only the local node if using shared binaries.
Enter the same ORACLE_HOME information used for the installation
Run post-install tasks

Run Veritas’ postinstallora script:

- root# cd /opt/VRTSvcs/rac/bin/script
- root# ./postinstallora
Are you ready to remove SFRAC ODM entries from the cluster [y/n/q]? Y
Answer “y” to proceed.

Verify whether GSD is running
- oracle$ gsdctl stat
  GSD is not running on the local node

If not, initialize SRVM location on one cluster node:
- oracle$ srvconfig -init –f

Start GSD daemon on both nodes:
- oracle$ gsdctl start
  Successfully started GSD on local node

**Configure service groups**
The example below sets up only auto-mount of filesystems. Be careful using this configuration, as the ‘hastop’ command will unmount these filesystems regardless of whether an Oracle database is still running on top of them.

See the Veritas Storage Foundation 4.0 for Oracle RAC – Installation and Configuration Guide (AIX) for details on configuring service groups to also start listener & database instances, if desired.

Stop VCS on each node:
- root# hastop –all –force
- root#/etc/VRTSvcs/conf/config=>hastatus -sum
  VCS WARNING V-16-1-11046 Local system not available

The service group configuration file is /etc/VRTSvcs/conf/config/main.cf. Make a copy of this file before beginning.
- root# cd /etc/VRTSvcs/conf/config
- root# cp main.cf main.old

**Add entries for all filesystems and mount points as shown in bold below.**

group cvm (  
  SystemList = { erpcc2 = 0, erpcc8 = 1 }  
  AutoFailOver = 0  
  Parallel = 1  
  AutoStartList = { erpcc2, erpcc8 }  
  )

  CFSfsckd vxfsckd (  
  )  
  CFSMount mt_orabin (  
  )
Critical = 0  
MountPoint = “/sfrac/oracle”  
BlockDevice = “/dev/vx/dsk/oradg/vol_orabin” 
)

CFSMount mt_oradata (  
  MountPoint = “/sfrac/oracle/oradata”  
  BlockDevice = “/dev/vx/dsk/oradg/vol_oradata” 
)

CVMVolDg ora_vols (  
  CVMDiskGroup = oradg  
  CVMVolume = { vol_orabin, vol_oradata }  
  CVMActivation = sw 
)

CVMCluster cvm_clus (  
  CVMClustName = rac_cluster  
  CVMNodeId = { erpcc2 = 0, erpcc8 = 1 }  
  CVMTransport = gab  
  CVMTimeout = 200 
)

CVMVxconfigd cvm_vxconfigd (  
  Critical = 0  
  CVMVxconfigdArgs = { syslog } 
)

cvm_clus requires cvm_vxconfigd  
vxfsckd requires cvm_clus  
mt_orabin requires ora_vols  
mt_orabin requires vxfsckd  
mt_oradata requires mt_orabin  
mt_oradata requires ora_vols  
ora_vols requires cvm_clus

Save and close main.cf

Verify syntax of main.cf file:
  • root# hacf -verify /etc/VRTSvcs/conf/config  
  • root# hacf -verify /etc/VRTSvcs/conf/config

From the node where main.cf was modified, join the cluster:
  • root# hastart
From the second node, join the cluster, specifying a stale configuration file. This will load in the changed configuration file from the other cluster node.

- root# hastart –stale

Check to see that cluster started:

- root# hastatus –sum

```
-- SYSTEM STATE
-- System     State     Frozen
A erpcc2     RUNNING    0
A erpcc8     RUNNING    0

-- GROUP STATE
-- Group       System     Probed  AutoDisabled  State
B ClusterService  erpcc2     Y       N           ONLINE
B ClusterService  erpcc8     Y       N           OFFLINE
B cvm         erpcc2     Y       N           ONLINE
B cvm         erpcc8     Y       N           ONLINE
```

**Set up demo database**

Most customers will have their own database create script to run; however, for the purposes of this document, we will configure a demo database and run Oracle’s Transparent Application Failover (TAF) demo to show that the setup is correct.

**Configure Listeners**

A RAC cluster requires both local and remote listeners. The listener.ora file and a name resolution method such as tnsnames.ora must both be set up prior to database creation, or the creation process will fail.

Create the file sqlnet.ora in $ORACLE_HOME/network/admin with the following contents:

```
NAMES.DEFAULT_DOMAIN = testlab.ibm.com

NAMES.DIRECTORY_PATH= (TNSNAMES)
```

Create the file listener.ora in $ORACLE_HOME/network/admin with the following contents:

```
SID_LIST_LISTENER_VRTS1 =
(SID_LIST =
 (SID_DESC =
 (SID_NAME = PLSExtProc)
```
(ORACLE_HOME = /sfrac/oracle/product/9.2.0)  
(PROGRAM = extproc)  
)  
(SID_DESC =  
(ORACLE_HOME = /sfrac/oracle/product/9.2.0)  
(SID_NAME = VRTS1)  
)  
)  
SID_LIST_LISTENER_VRTS2 =  
(SID_LIST =  
(SID_DESC =  
(SID_NAME = PLSExtProc)  
(ORACLE_HOME = /sfrac/oracle/product/9.2.0)  
(PROGRAM = extproc)  
)  
(SID_DESC =  
(ORACLE_HOME = /sfrac/oracle/product/9.2.0)  
(SID_NAME = VRTS2)  
)  
)  
LISTENER_VRTS2 =  
(DESCRIPTION_LIST =  
(DESCRIPTION =  
(ADDRESS_LIST =  
(ADDRESS = (PROTOCOL = TCP)(HOST = erpcc8)(PORT = 1521))  
)  
)  
)  
LISTENER_VRTS1 =  
(DESCRIPTION_LIST =  
(DESCRIPTION =  
(ADDRESS_LIST =  
(ADDRESS = (PROTOCOL = TCP)(HOST = erpcc2)(PORT = 1521))  
)  
)  
)  
Lastly, create the file tnsnames.ora with the following contents:

LISTENER_VRTS1 =  
(ADDRESS = (PROTOCOL = TCP)(HOST = erpcc2)(PORT = 1521))  

LISTENER_VRTS2 =  
(ADDRESS = (PROTOCOL = TCP)(HOST = erpcc8)(PORT = 1521))  

VRTS2 =  
(DESCRIPTOR =  
(ADDRESS_LIST =  
(ADDRESS = (PROTOCOL = TCP)(HOST = erpcc8)(PORT = 1521))  
)
LISTENERS_VRTS =
   (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = erpcc8)(PORT = 1521))
      (ADDRESS = (PROTOCOL = TCP)(HOST = erpcc2)(PORT = 1521))
   )
VRTS =
   (DESCRIPTION =
      (LOAD_BALANCE = YES)
      (ADDRESS_LIST =
         (ADDRESS = (PROTOCOL = TCP)(HOST = erpcc8)(PORT = 1521))
         (ADDRESS = (PROTOCOL = TCP)(HOST = erpcc2)(PORT = 1521))
      )
      (CONNECT_DATA =
         (SERVER = DEDICATED)
         (SERVICE_NAME = VRTS)
         (FAILOVER_MODE =
            (TYPE = SELECT)
            (METHOD = BASIC)
         )
      )
   )
VRTS1 =
   (DESCRIPTION =
      (ADDRESS_LIST =
         (ADDRESS = (PROTOCOL = TCP)(HOST = erpcc2)(PORT = 1521))
      )
      (CONNECT_DATA =
         (SERVICE_NAME = VRTS)
         (INSTANCE_NAME = VRTS1)
      )
   )

Verify that that configuration files are correct by starting and stopping the listeners on each cluster node:

- oracle$ lsnrctl start

LSNRCTL for IBM/AIX RISC System/6000: Version 9.2.0.4.0 - Production on 14-AUG-2004 13:40:19

Copyright (c) 1991, 2002, Oracle Corporation. All rights reserved.

Starting /sfrac/oracle/product/9.2.0/bin/tnslsnr: please wait...

TNSLSNR for IBM/AIX RISC System/6000: Version 9.2.0.4.0 - Production
System parameter file is /sfrac/oracle/product/9.2.0/network/admin/listener.ora
Log messages written to /sfrac/oracle/product/9.2.0/network/log/listener.log
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=erpcc2)
(PORT=1521)))

Connecting to (ADDRESS=(PROTOCOL=tcp)(PORT=1521))
STATUS of the LISTENER

-------------------------------
Alias         LISTENER
Version       TNSLSNR for IBM/AIX RISC System/6000: Version 9.2.0.4.
0 - Production
Start Date    14-AUG-2004 13:40:19
Uptime        0 days 0 hr. 0 min. 2 sec
Trace Level   off
Security       OFF
SNMP           OFF
Listener Parameter File   /sfrac/oracle/product/9.2.0/network/admin/listener.ora
Listener Log File         /sfrac/oracle/product/9.2.0/network/log/listener.log
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=erpcc2)(PORT=1521))
)
The listener supports no services
The command completed successfully

• oracle$ lsnrctl stop

LSNRCTL for IBM/AIX RISC System/6000: Version 9.2.0.4.0 - Production on 14-AUG-2
004 13:39:50
Copyright (c) 1991, 2002, Oracle Corporation. All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(PORT=1521))
The command completed successfully

Create demo database

• oracle$ dbca –datafileDestination /sfrac/oracle/oradata
If this screen does not list an “Oracle cluster database” as an option, Oracle did not detect that the system was in a cluster configuration. If so, cancel out of the utility and run the Veritas script /opt/VRTSvcs/rac/bin/script/preinstallora.
On this screen, unlike the Oracle installation screen, all cluster nodes should be selected regardless of whether shared Oracle binaries are used.
Specify the following database information:

- An Oracle database is uniquely identified by a Global Database Name, typically of the form "<database name>_domain".

**Global Database Name:** [Insert value]

A database is referenced by an Oracle instance on each cluster database node. Each instance is uniquely identified by an Oracle System Identifier (SID). For each cluster database instance, the SID is comprised of a common prefix for the database and a number for each instance that is automatically generated. A suggested SID prefix has been entered for you which you can accept or change to a value you prefer.

**SID Prefix:** [Insert value]

---

Listeners:

- Select the listener(s) for which you want to register this database.
- Register this database with all the listeners.
- Register this database with selected listener(s) only.

**Available Listeners:**

- LISTENER, VRT52
- LISTENER, VRT51

**Selected Listeners:**

- [Select/Unselect listener(s)]

---
Keep the defaults selected for this screen.
The default Shared Pool and Java Pool sizes are too small for a successful installation; increase to 200 MB

If this screen yields a file error, check the oracle user’s ability to rcp between cluster nodes. For demo purposes, use the default filenames.
Following database creation, SYS and SYSTEM passwords must be created.

**Transparent Application Failover (TAF) demo**

Complete instructions for installing the TAF demo code can be found at the following URL:
http://otn.oracle.com/sample_code/tech/java/sqlj_jdbc/files/9i_jdbc/OCIdriverTAFSample/Readme.html
Install Oracle 9i Client to Windows client machine
(client may be Windows 98, NT, 2000, or XP)

Download Oracle 9i to the client machine from the following URL:


unzip 92010NT_Disk1.zip and extract to c:\oracle\disk1
unzip 92010NT_Disk2.zip and extract to c:\oracle\disk2
unzip 92010NT_Disk3.zip and extract to c:\oracle\disk3

Launch c:\oracle\disk1\setup.exe
At a minimum, select Oracle Network Utilities, Oracle Database Utilities, Oracle Java Utilities, SQL*Plus, Oracle JDBC/OCI interfaces, and Oracle Call Interfaces.
Select “install” from the summary screen and complete the installation. If Oracle Enterprise Manager (OEM) was installed, it will launch automatically following the installation. Select File -> Exit to exit OEM.

**Install TAF Sample Code**

Download OCIdriverTAFSample.jar from [http://www.oracle.com/technology/sample_code/tech/java/sqljjdbc/files/9ijdbc/OCIdriverTAFSample/Readme.html](http://www.oracle.com/technology/sample_code/tech/java/sqljjdbc/files/9ijdbc/OCIdriverTAFSample/Readme.html) to c:\oracle\n
Unjar the file from dos prompt:

- `jar.exe xvf c:\oracle\OCIdriverTAFSample.jar`

Edit the document `c:\oracle\OCIdriverTAFSample\Connection.properties` as follows:

```properties
# Your Database Connection details
TNSEntryName = VRTS
UserName     = oe
Password     = oe
```

FTP the file `c:\oracle\OCIdriverTAFSample\database\OCIdriverTAFSample.sql` to the `/sfrac/oracle` directory on one of the cluster nodes and execute:

- `oracle$ sqlplus /nolog
SQL*Plus: Release 9.2.0.4.0 - Production on Fri Aug 6 10:31:25 2004
Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.
SQL> connect / as sysdba
SQL> alter user oe account unlock;
SQL> alter user oe identified by oe;
SQL> connect oe/oe
Connected.
SQL> @/sfrac/oracle/OCIdriverTAFSample.sql
```

**DOC>* @author Elangovan
**DOC>* @version 1.0
**DOC>* Development Environment : Oracle9i JDeveloper
**DOC>* Name of the Application : OCIdriverTAFSample.sql
**DOC>* Creation/Modification History :
**DOC>* Elangovan 10-Oct-2001 Created
**DOC>* Overview of Script :
```
This script performs the clean up and creates the tables required by the sample application. Check if the names used in this script are not used by other application.

Performing clean up

```
DROP TABLE Order_Data
```

ERROR at line 1:
ORA-00942: table or view does not exist

Creating Table Order_Data

Table created.

Commit complete.

End of Script

Add the following environment variables:

- C:\oracle> SET JAVA_HOME=c:\orahome9i\jdk
- C:\oracle> SET PATH=%JAVA_HOME%\bin;c:\orahome9i\bin;%PATH%
- C:\oracle> SET JDBC_HOME=c:\orahome9i\jdbc\lib
- C:\oracle> SET TNS_ADMIN=c:\orahome9i\network\admin

Run the application:

- C:\oracle> cd OCIdriverTAFSample
- C:\oracle\OCIdriverTAFSample> run.bat
Press the start button to begin the workload.

Determine which cluster node is running the workload by running the following SQL statement from each cluster node:

- `oracle$ sqlplus /nolog
  SQL> connect / as sysdba;
  Connected
  SQL> select distinct username from v$session;

<table>
<thead>
<tr>
<th>USERNAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>OE</td>
</tr>
<tr>
<td>SYS</td>
</tr>
<tr>
<td>SQL&gt;</td>
</tr>
</tbody>
</table>

The system with an “OE” username is running the load.

On this node, we will stop the database and observe TAF failover:

```
SQL> shutdown immediate
```
### Total No of Orders Placed

<table>
<thead>
<tr>
<th>Time</th>
<th>No of Orders</th>
</tr>
</thead>
<tbody>
<tr>
<td>13:11:14</td>
<td>105</td>
</tr>
<tr>
<td>13:11:21</td>
<td>105</td>
</tr>
<tr>
<td>13:11:27</td>
<td>105</td>
</tr>
<tr>
<td>13:11:34</td>
<td>105</td>
</tr>
<tr>
<td>13:11:40</td>
<td>105</td>
</tr>
</tbody>
</table>

- Failover has detected a lost connection and failover has started.
- Failover Type: SELECT failing over...
- Failover has ended successfully.
- Continuing with the Query ...
Appendix A: Differences between Storage Foundation for Oracle RAC 4.0 for AIX vs. Solaris

- For AIX, CFS & CVM are not available separately, they are packaged in VERITAS Storage Foundation 4.0
- For AIX, there is no "rootability" in VxVM – the rootvg volume group must be controlled by LVM
- Solaris SFRAC supports up to 16 nodes; AIX SFRAC initially supports up to 4 node, and is planned to allow up to 8 nodes by version 4.1
- Solaris SFRAC 4.0 doesn’t support PowerPath; AIX SFRAC 4.0 is planned to include it
- SCSI-2 disk reservation available on Solaris, not on HP or AIX (getting phased out in favor of SCSI-3 persistent reserve)
- Naming convention differences
Reminders

Copyright 2004 IBM Corporation. All Rights Reserved.

Neither this documentation nor any part of it may be copied or reproduced in any form or by any means or translated into another language, without the prior consent of the IBM Corporation. The information in this paper is provided by IBM on an "AS IS" basis. IBM makes no warranties or representations with respect to the content hereof and specifically disclaim any implied warranties of merchantability or fitness for any particular purpose. IBM assumes no responsibility for any errors that may appear in this document. The information contained in this document is subject to change without any notice. IBM reserves the right to make any such changes without obligation to notify any person of such revision or changes. IBM makes no commitment to keep the information contained herein up to date.

Version 1.0, published September 12, 2004
Trademarks

- IBM, AIX, and pSeries are trademarks or registered trademarks of the International Business Machines Corporation.
- VERITAS product names are trademarks or registered trademarks of VERITAS Software Corporation.
- Sun and Solaris are trademarks or registered trademarks of Sun Microsystems Company.
- Oracle, Oracle8, Oracle8i and Oracle9i are trademarks or registered trademarks of Oracle Corporation.
- UNIX is a registered trademark in the United States and other countries exclusively through X/Open Company Limited.
- All other products or company names are used for identification purposes only, and may be trademarks of their respective owners.
References

Nitin Vengurlekar, “Oracle Disk Manager” September 24, 2002

VERITAS Software Corporation “Easier Management with Oracle9i’s Oracle Disk Manager (ODM)” presentation

VERITAS Software Corporation “Implementing Manageable Real Application Clusters” presentation

VERITAS Software Corporation “Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide”

VERITAS Software Corporation “VERITAS Storage Foundation for Oracle RAC 4.0 Release Notes for AIX”

Oracle Corporation “Sample Illustrating Oracle9i JDBC OCI Transparent Application Callbacks;,
http://www.oracle.com/technology/sample_code/tech/java/sqlj_ocidbc/files/9i_ocidbc/OCIdriverTAFSample/Readme.html

Oracle Corporation “Step-By-Step Installation of 9i RAC on VERITAS Storage Foundation” Metalink Note ID 254815.1

IBM Corporation “Quick Installation Guide for Oracle9iR2/RAC on IBM pSeries on AIX 5.1 with HACMP”

Acknowledgements

Thanks to the following people for their assistance with details and configurations for this paper:

- John Fuchigami, Software Engineering Manager, VERITAS Software Corporation
- Jaymin Yon, Ralf Schmidt-Dannert, and Dale Martin, Advanced Technical Support for Enterprise Applications, IBM
- Ramesh Chitor, pSeries Solutions Development, IBM