NETAPP TECHNICAL REPORT

SETTING UP ORACLE® DATABASE 10*G*™ RELEASE 2 WITH REAL APPLICATION CLUSTERS WITH AUTOMATIC STORAGE MANAGEMENT OVER NFS ON NETAPP STORAGE

Niranjan Mohapatra, Network Appliance, Inc.

March 2007 | TR-3572

ABSTRACT

Oracle Database 10g RDBMS has a new facility called Automatic Storage Management (ASM) which provides an integrated cluster file system and volume management features. ASM complements the Oracle Database 10g RDBMS with both volume and disk management utilities, removing the need for third-party volume management tools while also reducing the complexity of the enterprise architecture. ASM provides simplicity of managing volumes that may be composed of block-based devices (SAN) or file-based devices (NFS). These devices are the underlying storage for the Oracle RDBMS.

Although ASM over SAN storage has been the more popular option, the customer has the choice of deploying ASM over NFS as well. The ASM layer works independently of the NFS client and therefore the NFS client resiliency and security are not compromised. ASM over NFS could be an option especially in the SMB segment where the TCO of deploying and managing a SAN might make that not a viable option. The storage administrator can follow a standard method of deploying and managing storage for structured as well as unstructured data. NFS is a ubiquitous protocol and therefore an Oracle DBA can easily manage the database requirements on NAS storage. NFS does not require personnel with the specialized skills that are required for administering a SAN environment. A customer who has an existing NS infrastructure can easily deploy an Oracle Database 10g Real Application Clusters (RAC) environment with ASM without any change in the storage environment.

This is also a feasible option for customers who are comfortable using NFS storage and wan to deploy Oracle Database 10g Standard Edition (SE). This license of the database does not have the tablespace partitioning feature. A good alternative for an NFS environment is spreading tablespaces across multiple volumes or storage using ASM disk-groups. The following links have more information about ASM and NetApp storage:

Oracle White Paper on ASM
Performance Comparison of Oracle RAC with Different Storage Protocols
ASM 10gR2 New Features
ASM 10gR1 Documentation
Oracle Database Administrator's Guide 10gR2 Chapter 12 "Using Automatic Storage Management "
Oracle Database Product Editions and Features

TABLE OF CONTENTS

1. INTRODUCTION

This technical report covers the installation of Oracle Database 10g Release 2 with Real Application Clusters (RAC) and Oracle Automatic Storage Management (ASM) over NFS on NetApp storage. The database is Oracle Real Application Clusters 10g Release 2 using Automatic Storage Management with Oracle Clusterware. This is now a certified configuration and therefore the components presented in this paper must be used in the same combination to gain support from all parties involved. The only exception is the application of certain patches as defined and required by all the vendors in this configuration. This report also covers the patches and recommendations for running Oracle Database 10g Release 2 with Real Application Cluster (RAC) and Automatic Storage Management (ASM) on NetApp Storage in an NFS environment.

2. ASSUMPTIONS

We assume that readers are familiar with Oracle Database10g Release 2 with Real Application Cluster (RAC) and with the operation of Network Appliance™ storage systems. We also assume that readers are familiar with the operation of the Enterprise Linux environment and installation of Oracle® Database 10g patches and any relevant Enterprise Linux rpms. It is also important to be familiar with all networking terminology and implementations.

3. SYSTEM ENVIRONMENT

The configuration presented in this document is based on the Oracle Database 10g Release 2 Real Application Cluster certification environment specified by Oracle and Network Appliance.

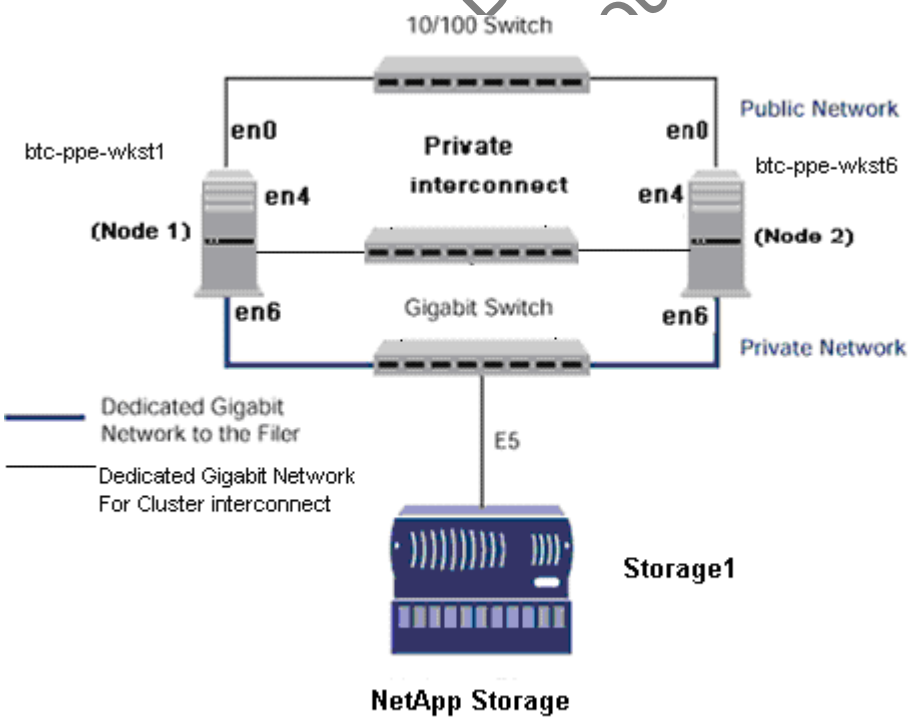


Figure 1) Oracle Database 10g Release 2 Real Application Cluster on IBM xSeries (Intel x86) servers with Network Appliance storage.

Figure 1 illustrates a typical configuration of Oracle Database 10g Release 2 Real Application Cluster with NetApp storage and IBM xSeries Intel servers running Oracle Enterprise Linux (OEL4). This scalable configuration enables users to scale horizontally and internally in terms of processor, memory, and storage.

As shown in Figure 1, NetApp strongly recommends that you dedicate a private network connection between the Oracle Database 10g Release 2 Real Application Cluster servers and the NetApp storage. This is accomplished by using a dedicated gigabit network (with a gigabit switch) to the NetApp storage. A dedicated network connection is beneficial for the following reasons:

- In an Oracle Database 10g Release 2 Real Application Cluster environment, it is important to eliminate any contentions and latencies.
- Providing a separate network ensures security.

The cluster interconnect is an essential part of Oracle Database 10g clusters. Along with cache fusion, it is also used to monitor the heartbeat of the servers in the existing cluster group. This is a typical configuration that can be deployed in a customer's environment.

4. REQUIREMENTS

4.1 HARDWARE

CLUSTER NODES

- Two IBM xSeries 306 32-bit Intel Servers
- One 4-Port 10/100 Base-TX Ethernet PCI Adapter
- One 10/100/1000 Base-T Ethernet PCI Adapter (for private interconnect)
- One 10/100/1000 Base-T Ethernet PCI Adapter (connected to NetApp storage)

STORAGE INFRASTRUCTURE

- One Network Appliance FAS2xx/F7xx/F8xx/FASF9x/FAS30xx system with Data ONTAP® 7.2 or later
- One gigabit switch with at least four ports.
- One gigabit NIC in the system
- One or more disk shelves, based on the disk space requirements

4.2 SOFTWARE

For both nodes in the participating cluster unless specified otherwise:

- Oracle Enterprise Linux 4
- Oracle Database 10g Release 2 (10.2.0.1), with Real Application Clusters license
- Oracle Database 10g Release 2 Patch 3.

5. SETUP FOR NETAPP STORAGE

For more information, refer to the Network Appliance installation and setup guides at http://now.netapp.com.

1. Configure a NetApp storage system running Data ONTAP 7.2 and with NFS and SnapRestore® license keys.

2. Create and export volumes for storing database files on the storage:

2.1. Create four volumes on the storage (Storage1) as listed below.

orahome Shared Oracle and CRS home (binaries)

oradata Oracle data files and control files

ora10g CRS files

oralogs Database logs, a copy of control file and archive logs.

To create volumes, use the following command at the NetApp storage console:

```
Storage1> vol create oradata 14
```

`Note`: We created volume oradata with 14 disks and volumes oralogs and orahome with 8 disks each. You can create your volumes based on your workload needs.

Add the following entries to the `/etc/exports` file on NetApp storage (Storage1):

```
/vol/orahome -anon=0
/vol/oradata -anon=0
/vol/oralogs -anon=0
/vol/ora10g  -anon=0
```

2.2. Execute the following command at the storage system console:

```
Storage1> exportfs -a
```


Note: NetApp recommends using flexible volumes in your database environment. NetApp FlexVol™ technology pools storage resources automatically and enables you to create multiple flexible volumes on a large pool of disks. This flexibility means that you can simplify operations, gain maximum spindle utilization and efficiency, and make changes quickly and seamlessly.

The database volume layout discussed in this document was defined for certification purposes, and your setup may vary depending upon requirements. For database layout best practices on NetApp storage, refer to NetApp Technical Report-3411 (www.netapp.com/library/tr/3411.pdf).

6. OPERATING SYSTEM CONFIGURATION

6.1 PATCHES

Before your Oracle Database 10g Release 2 Real Application Cluster installation, the following rpms need to be applied on IBM servers. Some of these rpms may already be applied to your system. Be sure to verify whether they already exist before applying them.

To determine whether the required rpms are already installed and committed, enter a command similar to the following:

```
# rpm –qa | grep compat
```

If a patch is not installed and committed, then install it. Here is a list of required patches.

binutils-2.15.92.0.2-21

compat-libstdc++-296-2.96-132.7.2

compat-db-4.1.25-9

compat-libstdc++-33-3.2.3-47.3

make-3.80-6.EL4

glibc-common-2.3.4-2.25

glibc-2.3.4-2.25

glibc-devel-2.3.4-2.25

openmotif-2.2.3-10.RHEL4.5

setarch-1.6-1

control-center-2.8.0-12.rhel4.5

gcc-3.4.6-3.1

gcc-c++-3.4.6-3.1

gnome-libs-1.4.1.2.90-44.2

libstdc++-3.4.6-3.1

libstdc++-devel-3.4.6-3.1

libaio-0.3.105-2

pdksh-5.2.14-30.3

Note: Java 1.4.2 32-bit is installed with Oracle.

6.2 OS SETTINGS

On Enterprise Linux systems, the default `ulimits` for individual users are set in `/etc/security/limits.conf`. As a root user, add the following entries using root users:

```
# Oracle specific settings
oracle soft nofile  4096
oracle hard nofile 65536
oracle soft nproc  2047
oracle hard nproc  16384
oracle soft memlock 3145728
oracle hard memlock 3145728
```

This needs to be done on all nodes of the cluster. . It may be necessary to reboot the server to activate updated limits. After you modify the settings, the ulimit `-a` command should display the following:

```
# ulimit -a

core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
file size               (blocks, -f) unlimited
max locked memory       (kbytes, -l) unlimited
max memory size         (kbytes, -m) unlimited
```

```
open files                      (-n) 4096
pipe size            (512 bytes, -p) 8
stack size              (kbytes, -s) unlimited
cpu time               (seconds, -t) unlimited
max user processes              (-u) 15168
virtual memory          (kbytes, -v) unlimited
```

Verify the above setting for the oracle user.

Kernel Settings

Add the following parameters for the shared memory and semaphores to the `/etc/sysctl.conf` file using root user.

```
kernel.shmall = 2097152
kernel.shmmax = 2147483648
kernel.shmmni = 4096
kernel.sem = 250 32000 100 128
fs.file-max = 65536
net.ipv4.ip_local_port_range = 1024 65000
net.core.rmem_default = 1048576
net.core.rmem_max = 1048576
net.core.wmem_default = 262144
net.core.wmem_max = 262144
```

Note: Set the parameter with the specified value in the `/etc/sysctl.conf` file:

vm.lower_zone_protection (= 100)

This parameter is to increases the default page threshold from 16MB to 100MB to ensure that kernel is available with sufficient memory during heavy OLTP operations on the system.

7. PREINSTALLATION SETUP TASKS (CLUSTER NODES)

This information is intended for both nodes in the participating cluster unless specified otherwise.

1. Have two IBM xSeries Intel servers ready with the latest recommended patches and OS settings as described in Section 6.

2. Install and configure NICs in the cluster nodes (three per node).

   2.1. Public IP: As indicated by name.

   2.2. Private interconnects: Connect one gigabit NIC to the gigabit switch of all the nodes for cluster interconnects.

   2.3. Server connection to NetApp storage: Connect one gigabit NIC to the gigabit switch, which will connect to the gigabit NIC on the NetApp storage.

3. Configure the network interfaces on each node.

   3.1. Configure the three network interfaces:

   `# btc-ppe-wkst1 (Host 1)`

```
en0 – ip: 10.73.68.155, netmask 255.255.254.0
en6 - ip: 10.73.69.155, netmask: 255.255.255.0
en4 - ip: 192.168.73.1, netmask: 255.255.255.0
# btc-ppe-wkst6 (Host 2)
en0 – ip: 10.73.68.156, netmask: 255.255.254.0
en6 - ip: 10.73.69.156, netmask: 255.255.255.0
en4 - ip: 192.168.73.2, netmask: 255.255.255.0
```

Where:

Interface en0 is the public IP for each node.

Interface en6 on both cluster nodes is connected to the gigabit switch for storage I/O.

Interface en4 on both cluster nodes is connected to the gigabit switch for cluster private interconnects.

3.2. Update the `/etc/hosts` file on the cluster nodes and add entries for public, private, and VIP addresses.

Note: In addition to the preconfigured public and private network, Oracle Database 10g requires additional IP addresses that will be mapped to the public address as virtual IPs (VIPs). If a node fails when an application or user makes a connection using a VIP, the Oracle Clusterware will transfers the VIP address to another surviving instance. You should add the VIP to the `/etc/hosts` file on all nodes in the cluster as well as all nodes that access the database. VIP must have the same subnet as the public IP address of the DB host.

An example of `/etc/hosts` entries:

```
# Internet Address Hostname # Comments
10.73.68.155  btc-ppe-wkst1      btc-ppe-wkst1.btcppe.netapp.com
10.73.69.155  btc-ppe-wkst1-en6
192.168.73.1  btc-ppe-wkst1-i     btc-ppe-wkst1-i.btcppe.netapp.com
10.73.68.195  btc-ppe-wkst1-v     btc-ppe-wkst1-v.btcppe.netapp.com
10.73.68.156  btc-ppe-wkst6       btc-ppe-wkst6.btcppe.netapp.com
10.73.69.156  btc-ppe-wkst6-en6
192.168.73.2  btc-ppe-wkst6-I      btc-ppe-wkst6-i.btcppe.netapp.com
10.73.68.196  btc-ppe-wkst6-v      btc-ppe-wkst6-v.btcppe.netapp.com
10.73.69.105  Storage1
```


4. Use the `ping` command to ensure the connectivity of each interface (interconnects, public IPs, and storage).

5. Create NFS mount points and mount the volumes with the following mount options on all the cluster nodes. As a root user, update the `/etc/fstab` file on all server nodes and add the following entries:

```
Storage1:/vol/oradata  /oradata      nfs
hard,nointr,proto=tcp,suid,vers=3,rw,bg,rsize=32768,wsize=32768,actimeo=0,timeo=600


Storage1:/vol/orahome  /orahome      nfs
hard,nointr,proto=tcp,suid,vers=3,rw,bg,rsize=32768,wsize=32768,actimeo=0,timeo=600


Storage1:/vol/oralogs       /oralogs        nfs
hard,nointr,proto=tcp,suid,vers=3,rw,bg,rsize=32768,wsize=32768,actimeo=0,timeo=600
```

```
Storage1:/vol/ora10g        /ora10g         nfs
hard,proto=tcp,suid,vers=3,nointr,rw,bg,rsize=32768,wsize=32768,noac,timeo=600
```

Where:

- Storage1 is the name of the NetApp storage system.
- oradata, oralogs, orahome, and ora10g are the mount points on the cluster nodes.
- ora10g is just a separate mount point for CRS files. CRS files (cluster registry file and voting disk file) can reside in the same `/ora10g` volume but they must be mounted with the `noac` mount option.
- During the Oracle CRS installation, be sure to indicate a path starting with the `/ora10g` directory when prompted for the cluster registry file (ocr) and voting disk (css) file location.

Note: : If you dynamically mount the NFS volumes without adding entries in the `/etc/fstab` file, the Oracle installation will fail.

Note: For details about the mount options, go to:

- http://now.netapp.com/Knowledgebase/solutionarea.asp?id=kb7518
- Oracle Metalink Note:359515.1

6. Create the following mount points on all cluster nodes:

```
#mkdir /oradata
#mkdir /oralogs
#mkdir /orahome
#mkdir /ora10g
```

Mount exported volumes on the mount points created above on all the cluster nodes. It is always a good idea to verify mount options by using the `mount` command on each node. After the NFS volumes are mounted, change the ownership of these mounted volumes to oracle user and dba group.

8. INSTALLATION PROCEDURE

The following sections explain the steps to install Oracle Database 10g Release 2 with Real Application Cluster on Enterprise Linux.

8.1. PREPARING TO INSTALL THE ORACLE REAL APPLICATION CLUSTER ON CLUSTERS NODES

1. This document assumes the Oracle user account and the group to be oracle and dba, respectively, on both cluster nodes. The user ID and group name for the oracle account should be the same on both cluster nodes. A sample oracle user `.bash_profile` file is provided in the Appendix A. Make sure that the user profile file exports at least the ORACLE_BASE, ORACLE_PRODUCT, ORACLE_HOME, ORACLE_SID, ORA_CRS_HOME, and PATH entries.

2. Grant appropriate permissions to the oracle user on all shared mounted volumes, /oradata, /orahome, and /oralogs:

```
#chown -R oracle:dba /oradata
#chmod -R 755 /oradata
```

Repeat this step for the orahome and oralogs volumes.

3. Set up account equivalence between the cluster nodes for the oracle user account. Add the following entries to the `/etc/hosts.equiv` file on all cluster nodes:

```
btc-ppe-wkst1 oracle
```

```
btc-ppe-wkst6 oracle
btc-ppe-wkst1-i oracle
btc-ppe-wkst6-i oracle
```

4. . Log in as the oracle user from both cluster nodes and then test the oracle account equivalence by using a remote shell utility such as `rsh`:

```
btc-ppe-wkst1:

#su - oracle
$rsh btc-ppe-wkst1 pwd
$rsh btc-ppe-wkst6 pwd
$rsh btc-ppe-wkst1-i pwd
$rsh btc-ppe-wkst6-i pwd

btc-ppe-wkst6:

#su - oracle
$rsh btc-ppe-wkst1 pwd
$rsh btc-ppe-wkst6 pwd
$rsh btc-ppe-wkst1-i pwd
$rsh btc-ppe-wkst6-i pwd
```


8.2. INSTALLING THE ORACLE REAL APPLICATION CLUSTER *10G* RELEASE 2 CLUSTER READY SERVICES (CRS)

For detailed information on installing Oracle Cluster Ready Services on Linux, refer to the *Oracle Real Application Clusters Installation and Configuration Guide 10*g *Release 2 (10.2.0.1) for LUNIX Systems* at http://otn.oracle.com/docs/content.html. This section briefly describes the procedures for using Oracle Universal Installer (OUI) to install Cluster Ready Services (CRS).

Note: The CRS home that you identify in this phase of the installation is only for CRS software; this home cannot be the same home as the Oracle Database 10g Real Application Cluster home. That is, ORACLE_HOME and CRS HOME must be different locations.

1. Run the `runInstaller` command from the `/crs` subdirectory on the Oracle Cluster Ready Services Release 2 (10.2.0.1) CD-ROM or from the staging area where Oracle CRS software has been dumped. This is a separate CD that contains the Cluster Ready Services software. This document assumes that OUI is started from node 1 (btc-ppe-wkst1). When OUI displays the Welcome page, click Next.

2. On the Specify Inventory page, enter a nonshared location for Oracle Inventory. This is the only part of Oracle Database 10g that should not be shared. For this test, we used `/home/oracle/oraInventory` for the Oracle Inventory information. Click Next.

3. The Specify File Locations page contains predetermined information for the source of the installation files and the target destination information. Specify the destination path for the shared CRS home. The path should be on a shared file system and different from $ORACLE_HOME. In this exercise, the shared CRS home was `/orahome/ora10g/product/10.2.0/crs_1`

4. On the next screen, specify the cluster name, public interface names (hostnames), private interface names and virtual interface hostnames to be used for the cluster interconnect. In our case, the public names

are btc-ppe-wkst1 and btc-ppe-wkst6 the private names are btc-ppe-wkst1-i and btc-ppe-wkst6-i and the virtual hostnames are btc-ppe-wkst1-v and btc-ppe-wkst6-v. Click next to continue.

5. On the Network Interface Usage page, specify the private network to be used for the cluster interconnect. This is a very important step. Do not leave it set to the default, which is Do Not Use. In our case, eth1 (btc-ppe-wkst1-i) was used as the private interconnect and eth0 (btc-ppe-wkst1) was used as the public interface. Select the interface and click the Edit button to modify it. Click Next.

6. On the Oracle Cluster Registry page, specify the OCR (Oracle Cluster Registry) file. Be sure to specify the full path to a shared location along with the name of the file. Do the same for a mirror file if you want normal redundancy. In our case, We used `/ora10g/ocrfile and /ora10g/ocrfile_mirror.` Click Next.

7. On the Voting Disk page, specify the CSS (Cluster Synchronization Services) voting disk file location. We used `/ora10g/cssfile` for `CSS` services. In case of normal redundancy specify the path along with name. Click Next to install the CRS.

8. When prompted run the following script as root user starting from primary node when it is prompted.

```
/orahome/ora10g/orainventory/orainstRoot.sh
```

```
/oarhome/ora10g/product/10.2.0/crs_1/root.sh
```

9. In the Configuration Assistant window you may see some warnings. Click OK to continue.

10. Run the `vipca` utility from the `$ORA_CRS_HOME/bin` directory as root user on the master Node (btc-ppe-wkst1). Click next.

11. Select the Public Interface. Click Next.

12. Specify the Virtual IP address and Subnet Mask of each node. Click Next.

13. Click Finish to continue vipca.

14. Click OK and then exit to finish VIPCA.

15. To verify your CRS installation by execute the `olsnodes` command from the `$CRS_HOME/bin` directory. The `olsnodes` command syntax is:

```
olsnodes [-n] [-l] [-v] [-g]
```

Where:

```
-n displays the member number with the member name
```

```
-l displays the local node name
```

```
-v activates verbose mode
```

```
-g activates logging
```

The output from this command should be a list of the nodes on which CRS was installed.

8.3. INSTALLING ORACLE DATABASE 10*G* RELEASE 2 SOFTWARE:

1. After making sure that Oracle Cluster Ready Services have started on the cluster nodes, start `runInstaller` from Disk1 of the Oracle Database10g Release 2 CDs or from the staging area where you have kept the Oracle Database 10g downloads.

2. On the Specify File Locations screen, enter the destination path for the shared ORACLE_HOME. This should be a different location than the shared CRS Home. For this exercise, the shared ORACLE_HOME was `/orahome/ora10g/product/10.2.0/db_1`.

3. On the next screen, select Cluster Installation and then select all the nodes in the cluster. For our exercise, the two cluster nodes were btc-ppe-wkst1 and btc-ppe-wkst6. Click Next.

Note: If the nodes are not displayed in the cluster node selection, then Oracle Cluster Ready Services are not configured or started on those cluster nodes.

4. For installation type, select Enterprise Edition and click Next.

5. On the Select Database Configuration screen, select "Do not create a starter database" . We used `dbca` to create a database later. Click Next.

6. Run the following scripts as root user starting from master node when prompted.

```
./$ORACLE_HOME/root.sh
```

7. Click exit to finish the database installation.

Note: Install Oracle Database 10g Release2 Patch 3 on both CRS_HOME and ORACLE_HOME using OUI Oracle Universal Installer (OUI). For more details about the patch installation, refer to the Oracle Patch Installation Guide.

9. CONFIGURING A LISTENER FOR AN ASM INSTANCE

1. Run `netca` (Oracle Net Configuration Assistant) as oracle user to create a listener.
2. Select Cluster Configuration and click Next to select all the nodes displayed to create the listener.
3. Select the default listener name that is displayed.

10. CREATING AN ASM INSTANCE

Note: The ASM instance created manages the Oracle data on the NFS file system instead of on raw devices. Therefore the use of ASMLib is not required.

To create the ASM instance, you will need device file or block partitions that Oracle will use to disk groups. Before creating an ASM instance, create device files to act as ASM disks, using the following commands as oracle user:

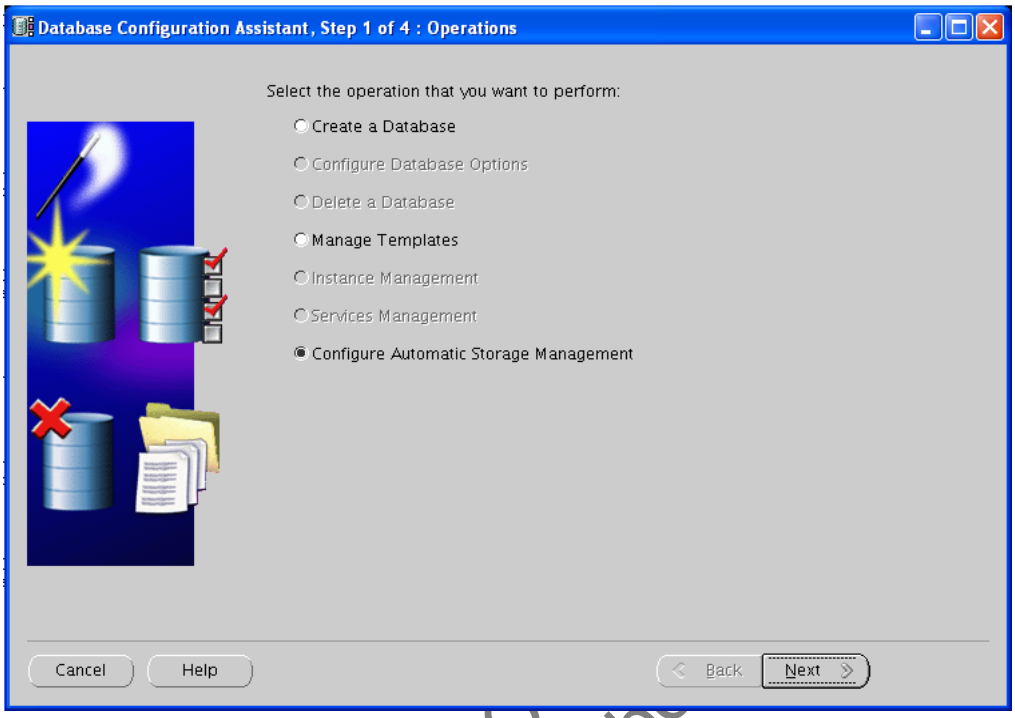
```
dd if=/dev/zero of=/oradata/_file_disk1 bs=1M count=30720
dd if=/dev/zero of=/oradata/_file_disk2 bs=1M count=30720
```

Note: You can create multiple volumes in NetApp storage for the database, according to your business requirements.

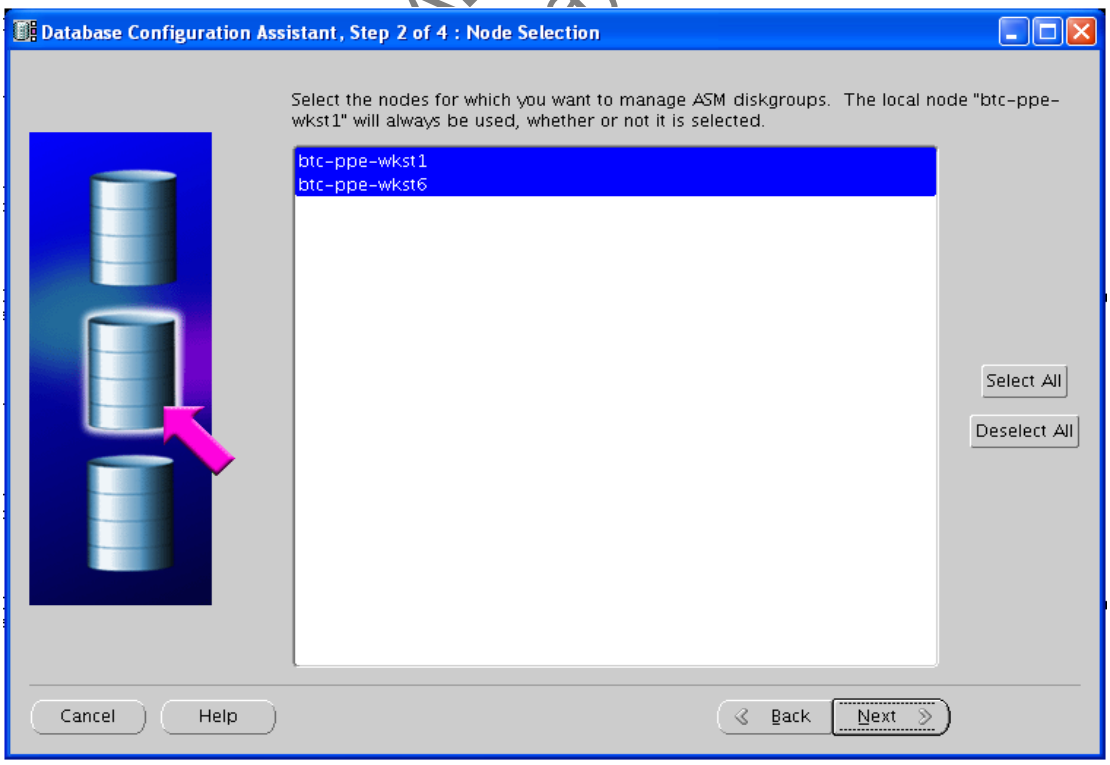
Follow these steps to create an ASM instance:

1. Start Database Configuration Assistant (DBCA) as oracle user and select Oracle Real Application Clusters Instance.

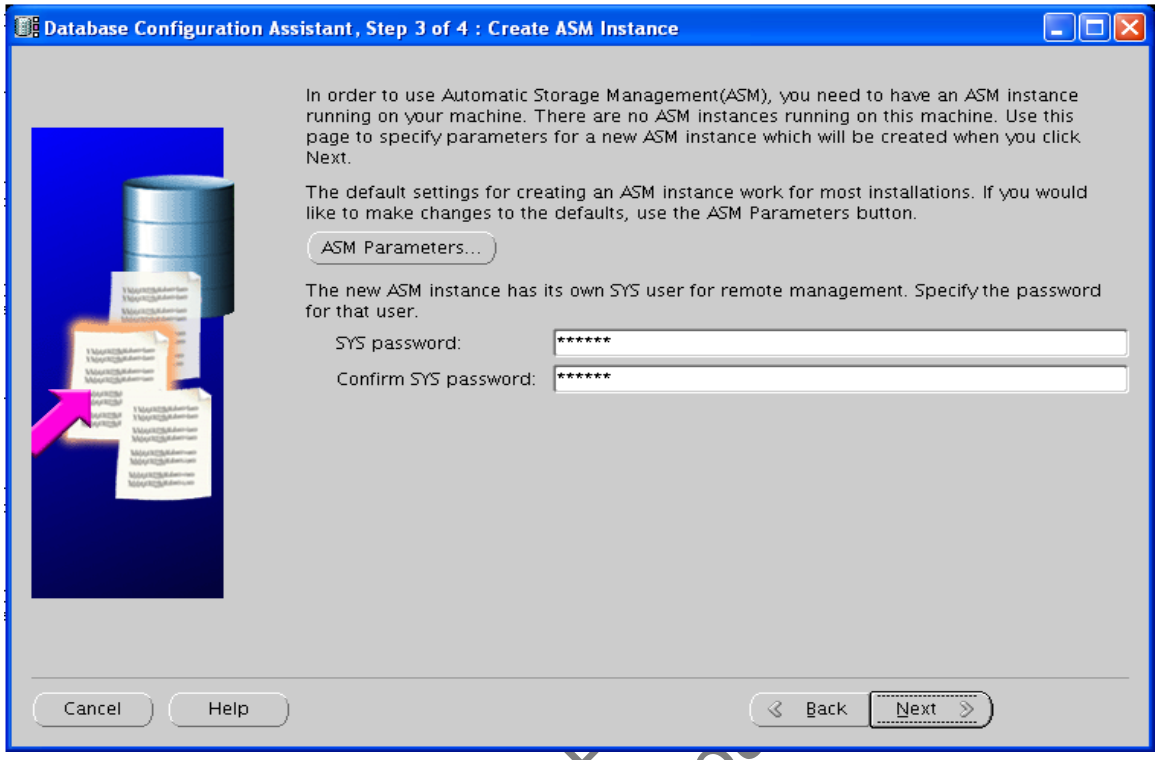
2. In the Operations screen, select Configure Automatic Storage Management and click Next.



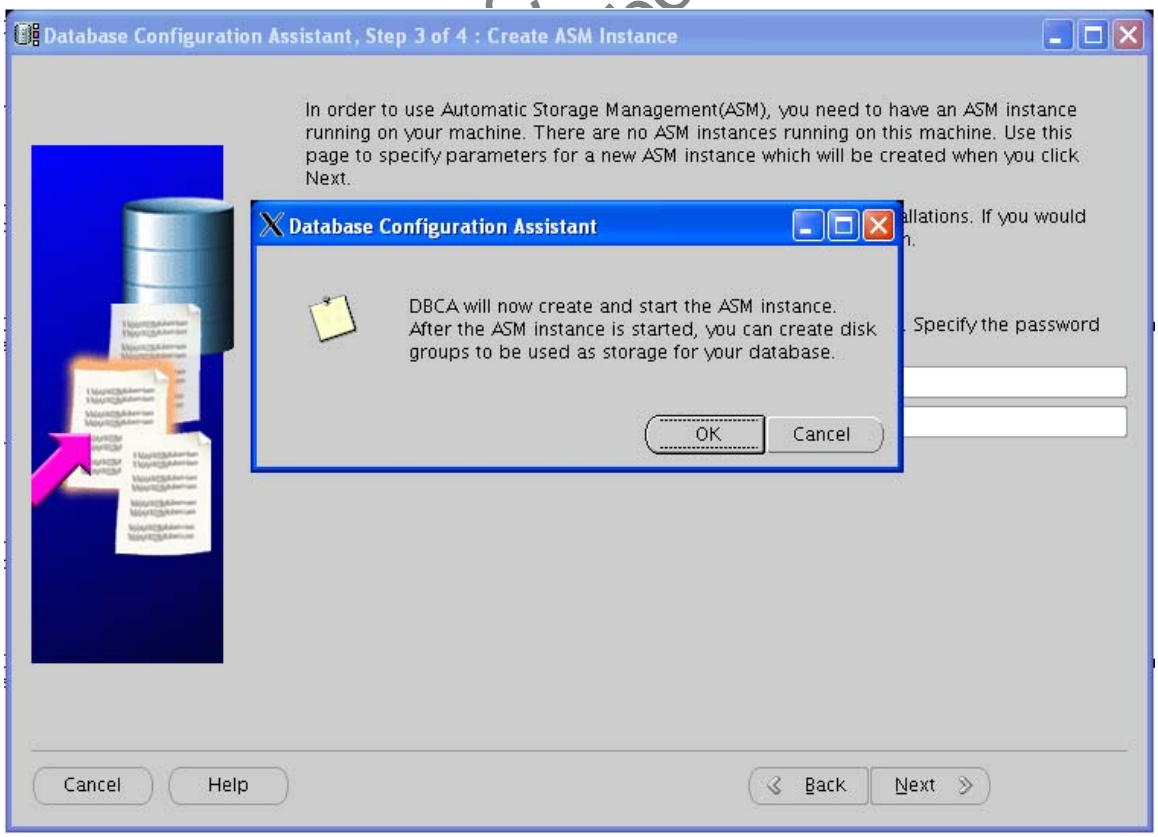
3. In the Node Selection screen, select all the nodes and click Next.



4. In the Create ASM Instance screen, enter the password for the SYS user and click Next.



5. DBCA prompts you to start the creation of ASM instance. Click OK to continue.



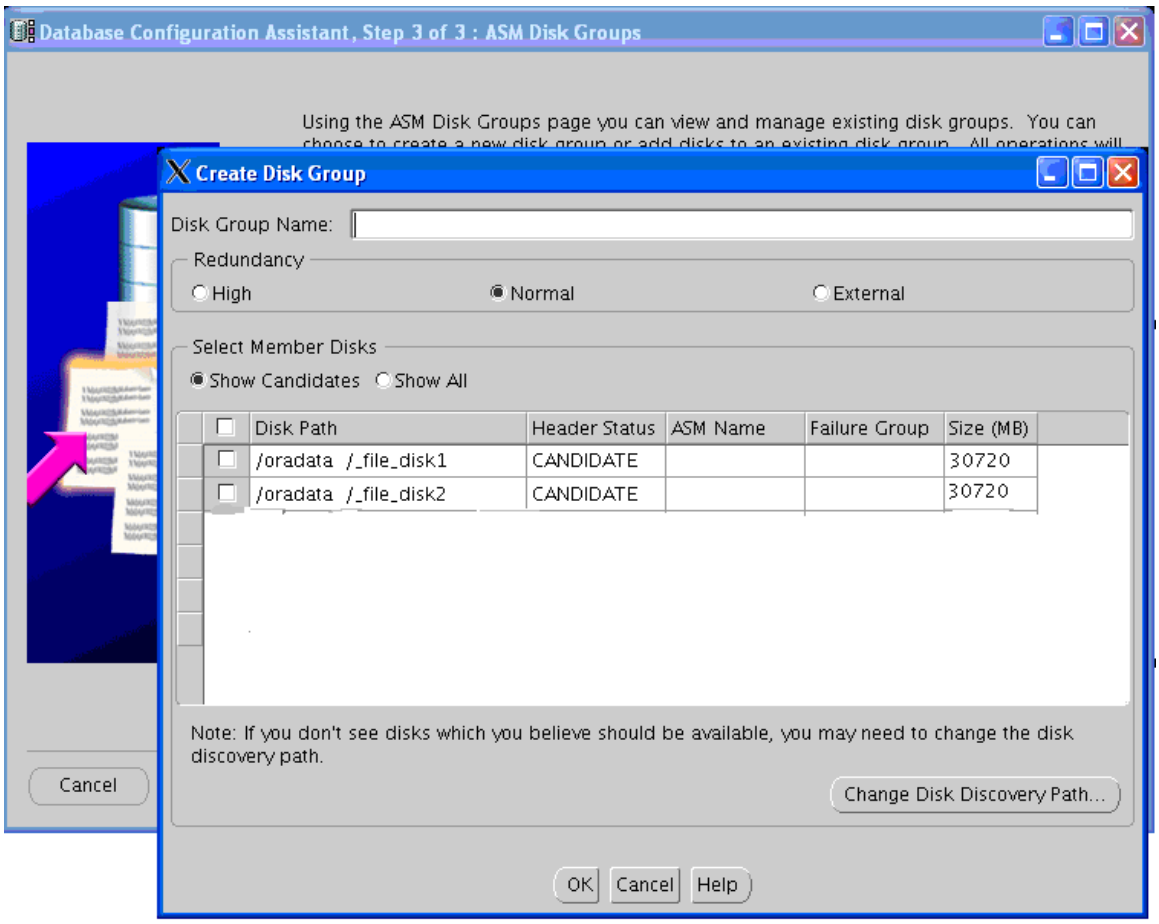
6. After the instance is created, DBCA prompts you to create a disk group. Click Create New to create a disk group. Click Change Disk Discovery Path to change the disk device location. Provide the following disk string path:

```
/oradata/*
```

Note: If you have Multiple volumes on NetApp Storage that you have mounted on the servers to store the database, you can specify those mounted directories in the disk string path putting a "-,-" as separator.

Example: `/oradata1/*, /oradata2/*, /oradata3/*, and so on.`

All the disks become visible.

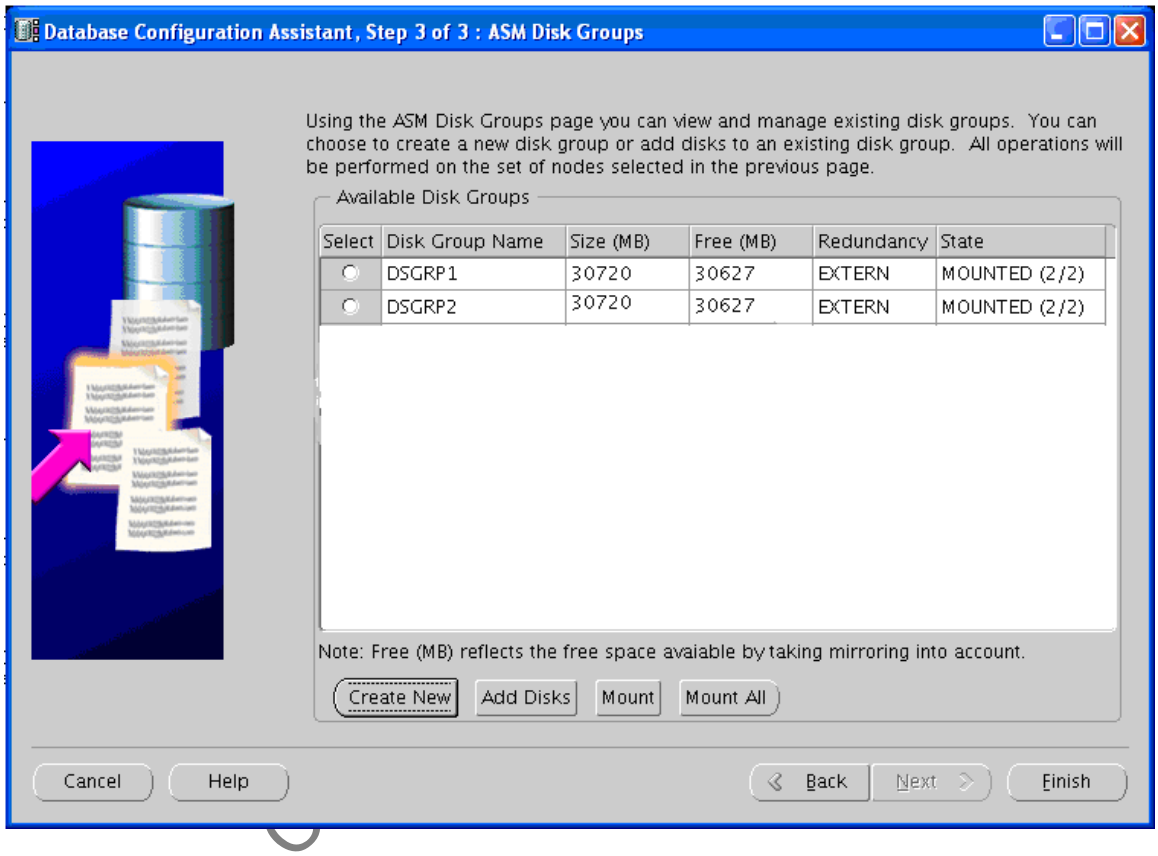


7. Enter a disk group name and select a redundancy option. To learn about redundancy go to **http://www.oracle.com/technology/products/manageability/database/pdf/asmwp.pdf**.

Select the disks to include in the disk group and click OK. DBCA creates the disk group and mounts it on the node from where DBCA is running. Then it mounts the disk group on other nodes of the cluster.

Note: If the ASM instance is not registered with the listener then DBCA returns a listener connectivity error when trying to mount the disk group on the other node. In that case, you can login in to the respective ASM instance and manually mount the disk group from the SQLPlus prompt using the "`alter diskgroup dsgrp1 mount`"; command.

8. Repeat step 7 to create the required disk groups.



9. When all disk groups have been created, click Finish to exit DBCA.

Note: If DBCA does not mount the disk group on the other ASM instance, logon to the ASM instance as user SYS on the other node and check the status of the disk groups:

```
Select name, state from v$asm_diskgroup;
```

The output of this command will shows the status (mounted or unmounted) of the disk groups. If the status of the disk groups is unmounted, then execute the following command to mount the disk groups:

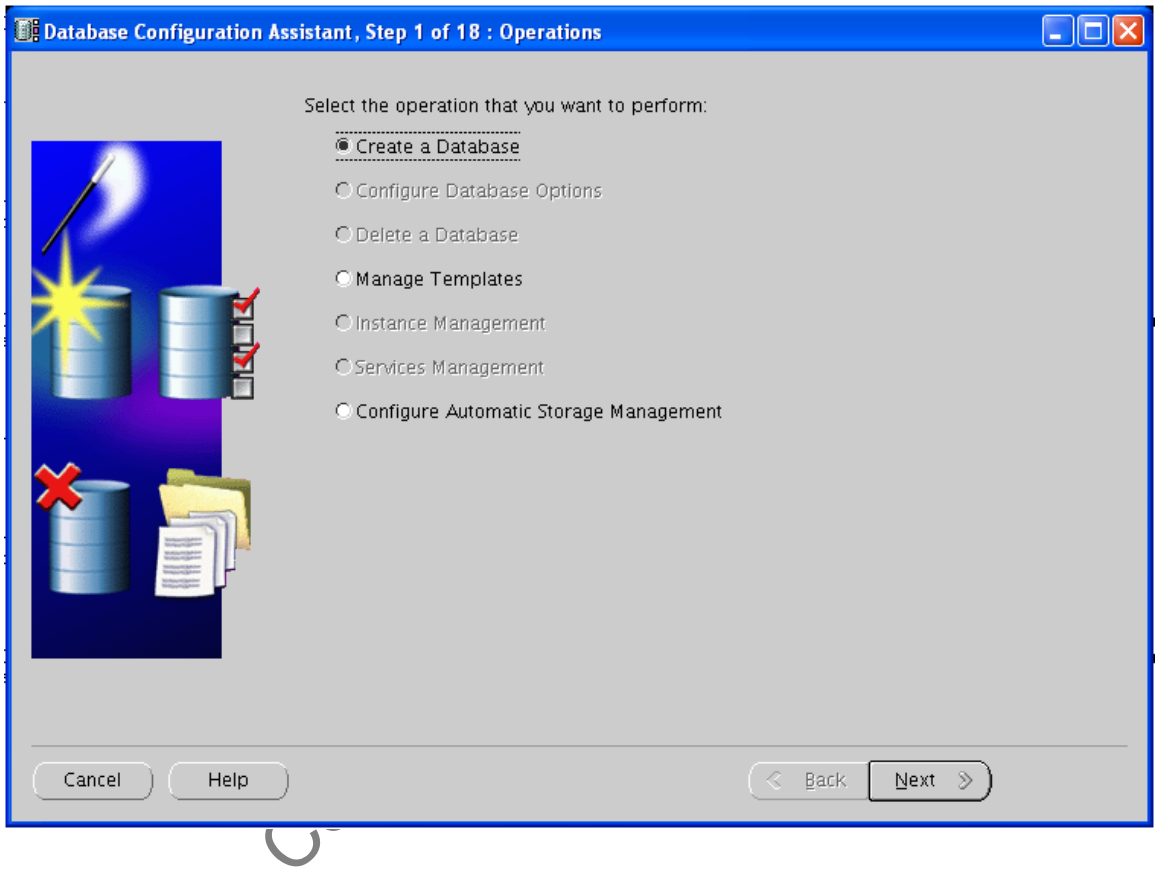
```
Alter disk group <diskgroup_name> mount;
```

11. CREATING A DATABASE USING ASM DISK GROUPS

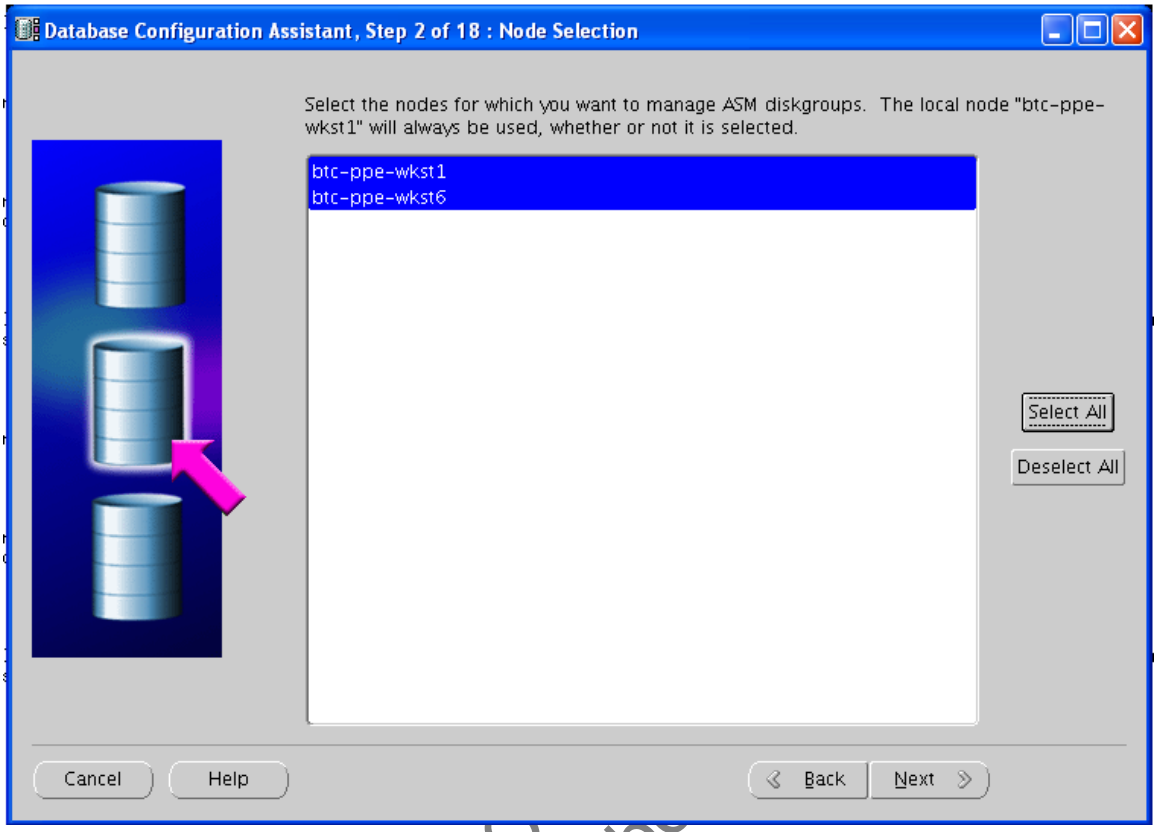
ASM disk groups can be used to store all types of database files, including binary and test files. We chose ASM disk group to store data files, control files, online redo logs, flash recovery area, and parameter files. We kept archive log and trace file destinations on file system.

Follow these steps to create an Oracle Database using ASM disk groups:

1. Start DBCA and select Create a Database.



2. In the Node Selection screen, select all the nodes and click Next.



3. In the Database Templates screen, select General Purpose and click Next.

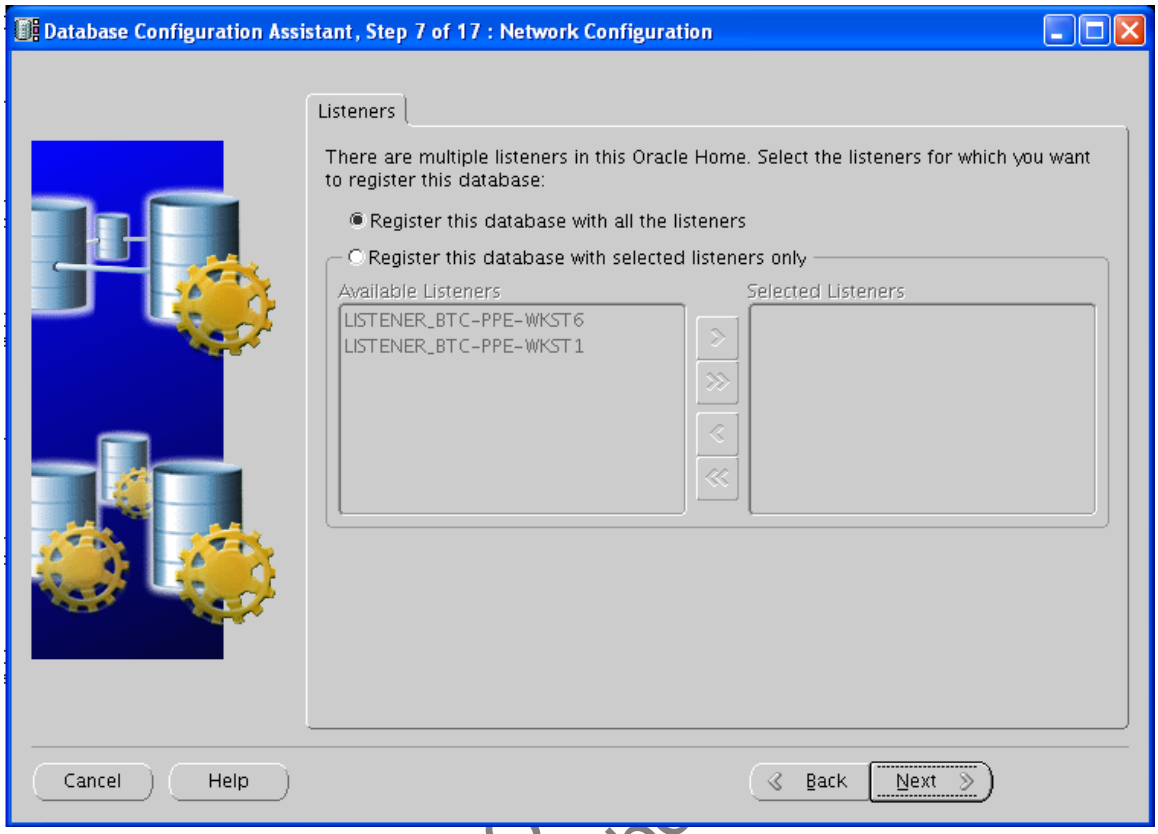
4. In the Database Identification screen, enter a Global Database Name and SID prefix. Click Next.

5. In the Management options, select Configure the Database with Enterprise Manager to configure the DB console and its repository. Click Next.

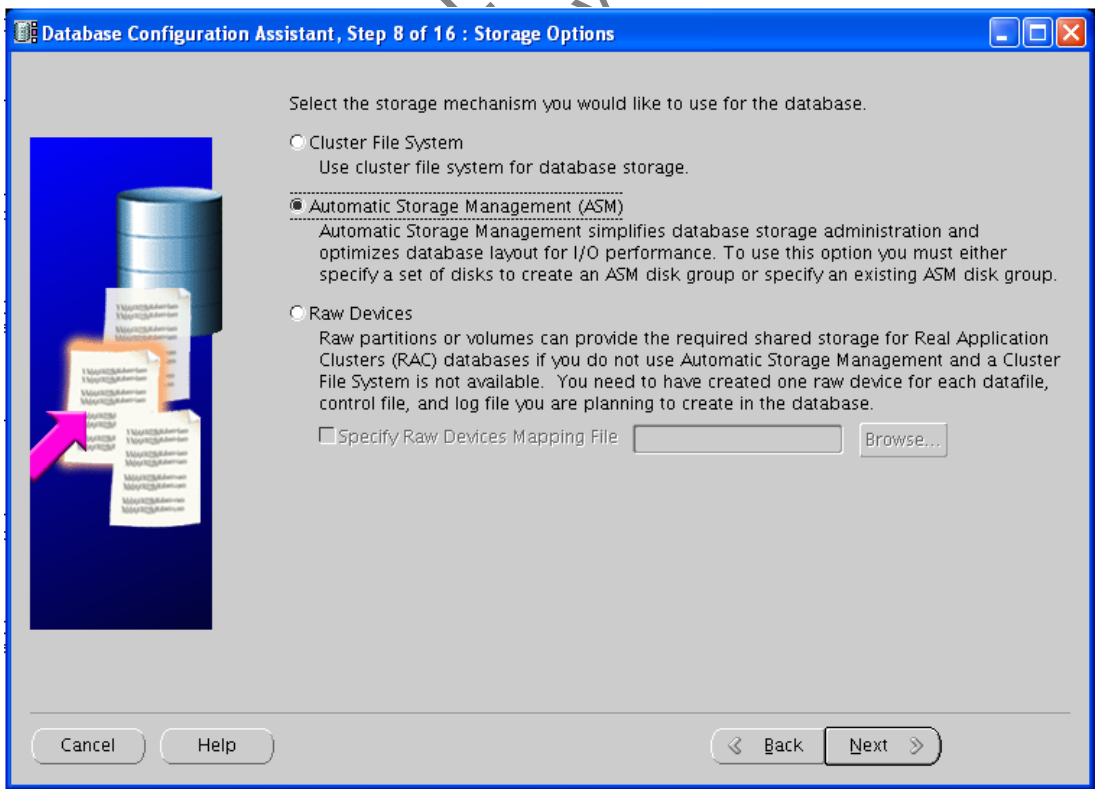
6. In the Database Credentials screen, enter a password for SYS and SYSTEM and click Next.

Note: If you have already created a RAC listener, DBCA prompts you to register the database with the listener in the Network Configuration screen. If you want to create separate listener for database, then skip this option.

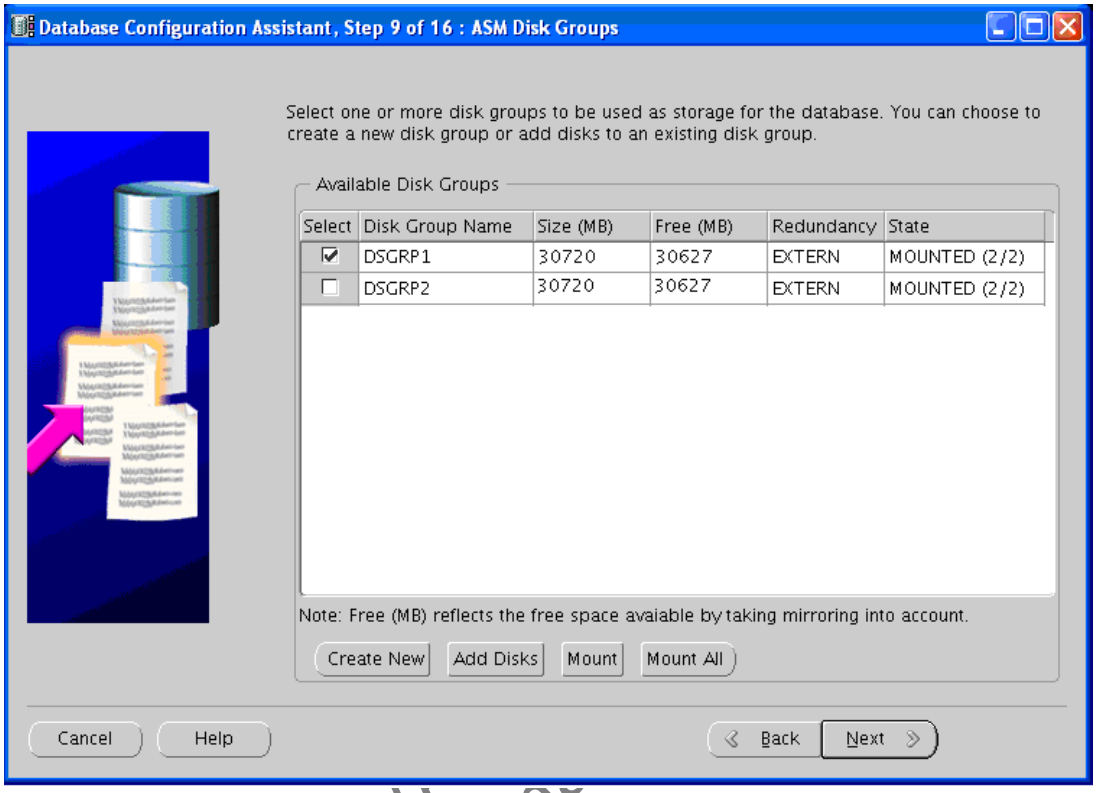
7. In the Network Configuration screen, select "Register this database with all listeners" and click Next.



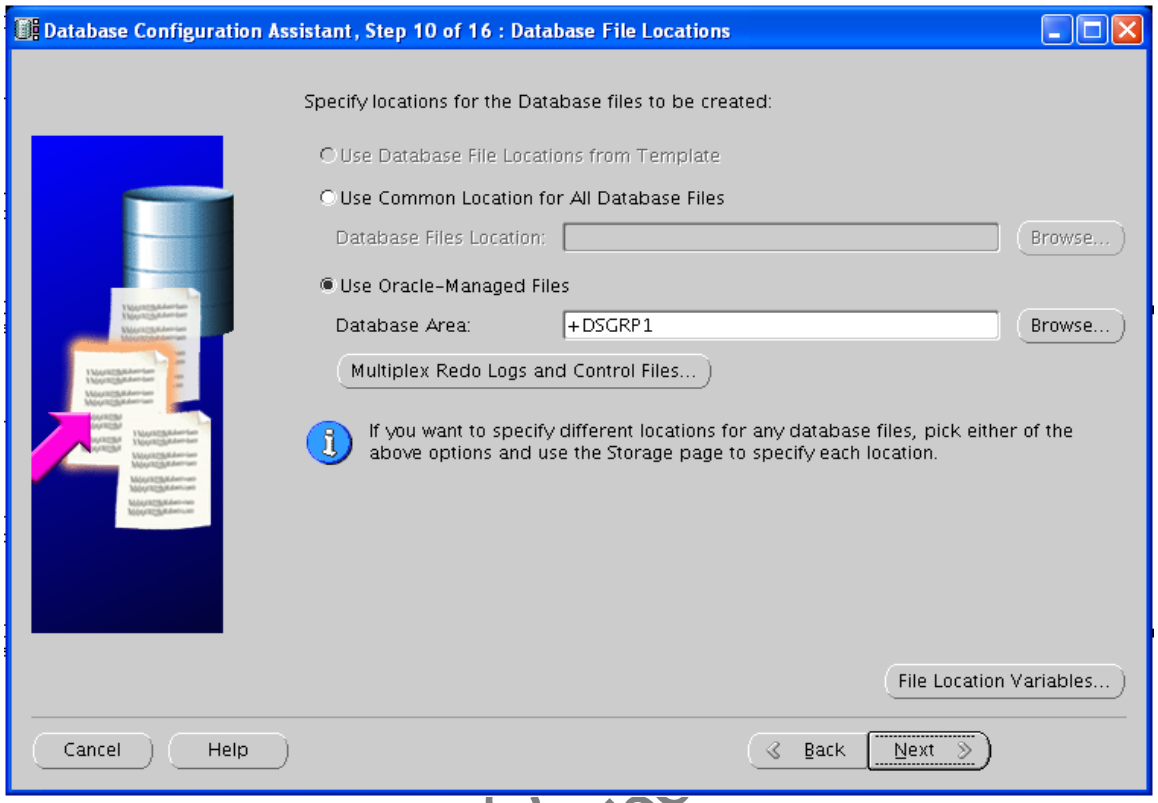
8. In the Storage Options screen, select option "Automatic Storage Management (ASM) and click Next.



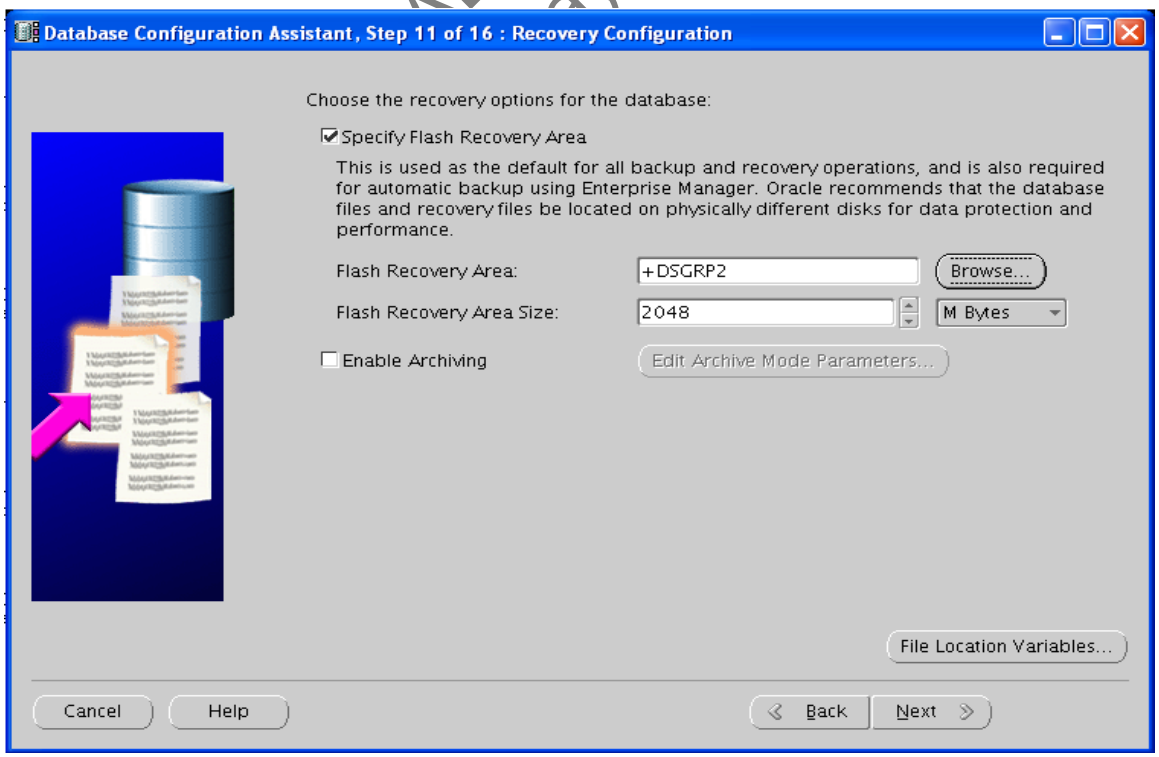
9. In the ASM Disk Groups screen, select the disk groups where you want to create or store the database files. In this example, disk group DSGRP1 is selected to store system files, control files and online redo log files. Click Next.



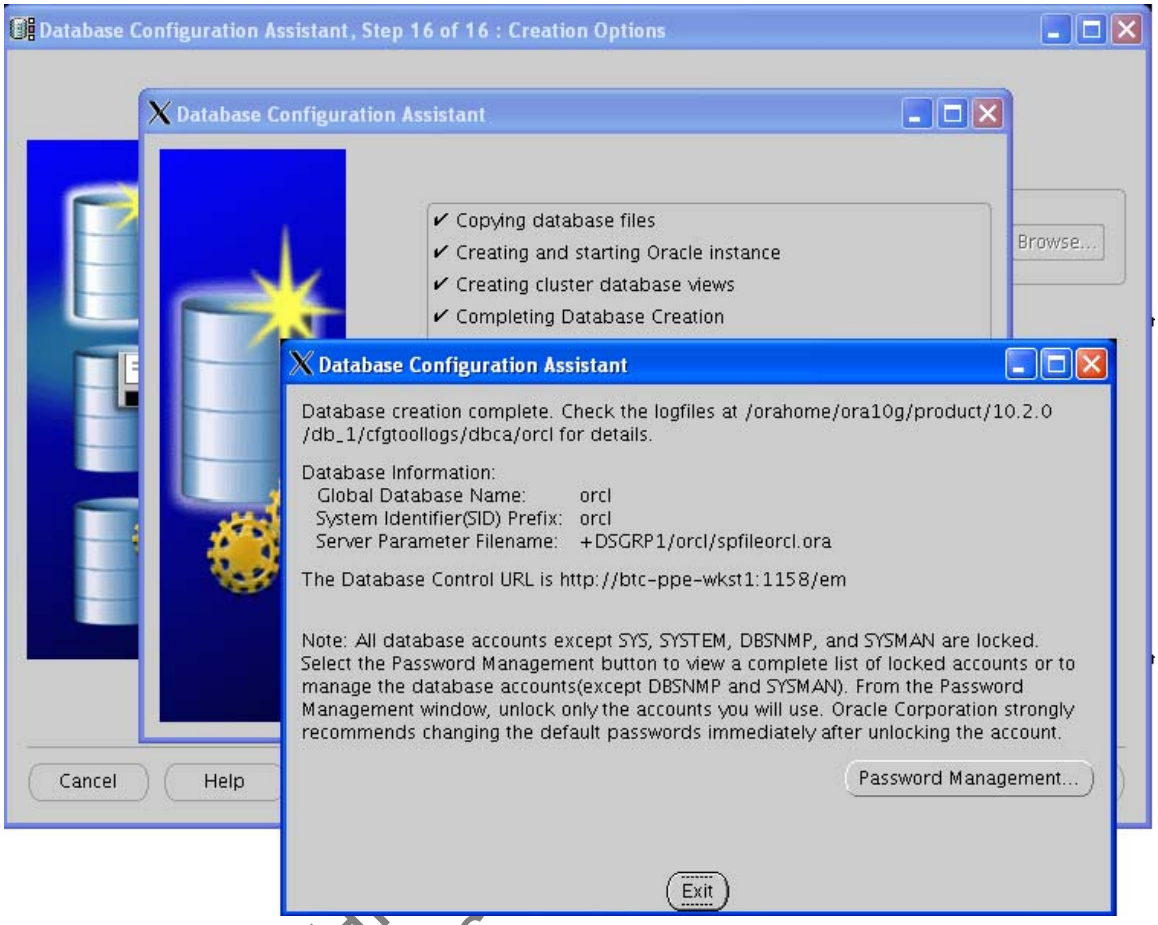
10. In the Database File Locations screen, select an option. In this test process, we selected “Use Oracle-Managed files.” Enter the disk group in which DBCA will create all the database files. Click Next.



11. In the Recovery Configuration screen, select Specify Flash Recovery Area. Enter disk group DSGRP2 as the flash recovery area destination and enter 2048 as the size. Click Next.



12. In the next screen, accept the default options. In the Creation Options screen, click Finish to start the database creation. After successful creation of the database, DBCA shows the completion message.



12. DATABASE SETTINGS FOR IO PERFORMANCE

Logon as oracle user, export the ORACLE_SID and connect to the database instance as the SYS user. Check the parameter settings in spfile as follows:

```
[root@btc-srv1 ~]# su - oracle

[oracle@btc-srv1 ~]$ export ORACLE_SID=orcl1

[oracle@btc-srv1 ~]$ sqlplus "/ as sysdba"
```

Check for the `disk_asynch_io` and `filesystemio_options` parameter settings in spfile:

```
SQL > show parameter disk_asynch_io
```

```
NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
disk_asynch_io                       boolean     FALSE
```

If `disk_asynch_io` is FALSE , change this parameter value to TRUE with the following command:

```
SQL > alter system set disk_asynch_io=TRUE scope=spfile;

SQL > show parameter filesystemio_options

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
filesystemio_options                 string      NONE
```

If the `filesystemio_options` setting is NONE, change this parameter value to `setall` with the following command:

```
SQL > alter system set filesystemio_options = setall scope=spfile;
```

Bring down all the database instances and ASM instance on all the cluster nodes.

Relink the oracle binaries in all cluster nodes as follows:

```
 [oracle@btc-srv1 ~]$ cd $ORACLE_HOME/rdbms/lib

[oracle@btc-srv1 ~]$ make PL_ORALIBS=-laio -f ins_rdbms.mk async_on
```

Repeat this command in all cluster nodes: It should relink the Oracle binaries without any error.

Start_up the ASM instance and database instance on all cluster nodes.

To confirm whether both direct I/O and asynch I/O are enabled, issue the following commands as oracle user:

```
[oracle@btc-srv1 ~]$ cat  /proc/slabinfo | grep kio


kioctx          53   75   256   15   1 : tunables  120   60   8 : slabdata    5    5    0
kiocb           47   64   128   31   1 : tunables  120   60   8 : slabdata    3    5    0
```

`Non-zero` values indicate that both direct I/O and asynch I/O are enabled.

13. APPENDIXA: SAMPLE .BASH_PROFILE FILE FOR THE ORACLE USER

```
export ORACLE_BASE=/orahome/ora10g;
export ORACLE_PRODUCT=$ORACLE_BASE/product;
export ORACLE_HOME=$ORACLE_PRODUCT/10.2.0/db_1;
export ORACLE_CRS=$ORACLE_PRODUCT/10.2.0/crs_1;
export ORACLE_SID=orcl;
export
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$ORACLE_CRS/lib:$ORACLE_HOME/lib32:$LD_LIBRAR
Y_PATH;
export LIBPATH=$ORACLE_HOME/lib:$ORACLE_CRS/lib:$ORACLE_HOME/lib32:$LIBPATH
export PATH=$PATH:$ORACLE_HOME/bin:$ORACLE_HOME:$ORACLE_CRS/bin:/usr/java14/bin;
```

14. ACKNOWLEDGEMENTS

The author would like to thank the following individuals for their contribution to the certification process and to this technical report:

Daniel Morgan, Shanthi Adloori, Uday Shet, Vasu Subbiah, Network Appliance Inc.

15. DISCLAIMER

Each environment has its own specific set of requirements and no guarantees can be given that the results presented in this report will work as expected on other platforms. This paper should assist in the research and troubleshooting that may be required in a particular case and serve as a checklist of items to be aware of. Please send any errors, omissions, differences, new discoveries, and comments about this paper to niranjan.mohapatra@netapp.com