Bookmark Go to End

Step-By-Step Installation of RAC on HP-UX

Purpose

This document provides the reader with step-by-step instructions on how to configure a cluster, install Oracle Real Application Clusters (RAC) and start a cluster database on HP-UX. For additional explanation or information on any of these steps, please see the references listed at the end of this document.

Disclaimer: If there are any errors or issues prior to step 3.3, please contact HP Support. The information contained here is as accurate as possible at the time of writing.

1. Configuring the Clusters Hardware

1.1 Minimal Hardware list / System Requirements

For a two node cluster the following would be a minimum recommended hardware list.
Check the RAC/HP certification matrix for RAC updates on currently supported hardware/software.

1.1.1 Hardware

All hardware is provided by HP. A bridge or hub is mandatory when using ethernet for the

interconnect. HP does not support a straight through ethernet connection for a private ethernet interconnect.

- For HP servers, third-party storage products, Cluster interconnects, Public networks, Switch options, Memory, Swap
  & CPU requirements consult the operating system or hardware vendor and see the RAC/HP certification matrix.

- Private network connection can be any of the following (see the RAC/HP certification matrix):
  HyperFabric
  RS232 Serial
  Ethernet or FDDI

- If the cluster is a 2 node configuration, any of the available disk subsystems provided by HP or their vendors are supported. Each of the disk arrays are dual-ported to each node, meaning that each array has a cable directly connecting it to each node.

- Disk Subsystem Options (see the RAC/HP certification matrix):
  2-4 nodes:
  A3311A Deskside fast/wide
  A3312A Rackmount fast/wide
  2-8 nodes:
  Fibre Channel such as EMC Symmetrix

- RAM Memory allocation: Minimum 256 MB. Use the following command to verify the amount of memory installed on your system
  $ /usr/sbin/dmesg | grep "Physical:"

- Swap Space: Minimum 2 x RAM or 750 MB, whichever is greater. Use the following command to determine the amount of swap space installed on your system:
  $ /usr/sbin/swapinfo -a (requires root privileges)

1.1.2 Software

- OS version >11.0 64bit

- ServiceGuard OPS Edition >11.09 Confirm with HP what version of ServiceGuard is certified with the specific OS version.

- Oracle Version 9.x is a 64bit Application and is available for HP 64bit Operating systems only. Verify that the OS is running a 64bit Kernel:

  #/bin/getconf KERNEL_BITS

- Use SAM and ensure that the following OS Kernel Parameter Settings are set to a minimum of these values:

  MAXDSIZ 536870912 bytes
  MAXDSIZ_64BIT 2147483648 bytes
  SEMMNS 2000
  SEMVMX 32768
  SHMMAX 1073741824 bytes or Available physical memory
  SHMMNI 512
  SHMSEG 32

SHMMIN 1
SHMMNI 100
SHMSEG 10
SEMMNI 70

- When using HMP (Hyper Messaging Protocol) instead of UDP for the interconnet protocol, the file /opt/clic/lib/skgxp/skclic.conf contains HMP configuration parameters that are relevant to Oracle

CLIC_ATTR_APPL_MAX_PROCS = Maximum number of Oracle processes.

CLIC_ATTR_APPL_MAX_NQS = Being obsoleted. Set to the same as CLIC_ATTR_APPL_MAX_PROCS

CLIC_ATTR_APPL_MAX_MEM_EPTS = 5000 Maximum number of Buffer descriptors.

CLIC_ATTR_APPL_MAX_RECV_EPTS = Maximum number of Oracle Ports = CLIC_ATTR_APPL_MAX_PROCS

CLIC_ATTR_APPL_DEFLT_PROC_SENDS = 1024 Maximum number of outstanding sends

CLIC_ATTR_APPL_DEFLT_NQ_RECVS = 1024 Maximum number of outsdanding receives on a port

- If the HMP cannot be used UDP may be sufficient over both HyperFabric cards or Ethernet cards where HMP can only be used over the HyperFabric interface. There are no addition settings when using UDP protocol.

1.1.3 Patches

Ensure that you are running the lasted Patch release. Patches listed will be superceded by new patches. Oracle always recommends to have the most current patch set from the OS vendor. You can also reference Document id 43507.1 in Oracle Metalink which is a best effort to update customers on OS patches that are needed.

HP provides patch bundles at

http://www.software.hp.com/SUPPORT_PLUS

Individual patches can be downloaded from

http://itresourcecenter.hp.com

To determine which operating system patches are installed, enter the following command:

$ /usr/sbin/swlist -l patch

To determine which operating system bundles are installed, enter the following command:

$ /usr/sbin/swlist -l bundle

1.2 Installing Disk Arrays

Prior to installing the operating environment and Cluster software, ensure that the shared disks are installed and available to each node (see the RAC/HP certification matrix). Typical cabinet or disk enclosures have disks preinstalled. Simply connect the cabinet to each server SCSI controller via standard SCSI connections. There are special consideration for A3311A and the A3312A disk subsystems. Both the A3311A and A3312A use simple SCSI connections, hence it is essential that all devices on a single SCSI chain have a unique target ID. This ID not only includes all the disks in the A3311A and A3312A but also the two hosts SCSI controllers to which the SCSI cables are connected from the A3311A. Change one of the nodes SCSI ID address from the default 6 to 7 on the SCSI controllers based on the hardware guide. This is only an issue when SCSI is being used.

1.3 Installing Cluster Interconnect and Public Network Hardware

For the Network configuration, have at least 2 connections to send a heartbeat from one system to the other (Ethernet, FDDI, RS232, Fibre Channel). That means that the minimal recommended configuration would consist of the following:

1 x Ethernet for Data, cluster interconnect heartbeat of the Cluster
1 x RS232 connection for the cluster heartbeat

Another configuration would be:

1 x Ethernet for Data, cluster interconnect heartbeat of the Cluster
1 x Ethernet as Standby network (that means no IP address!) - This would be configured as a second ip address subnet.

NOTE: Both networks must be bridged (recommendation from HP)

If you have configured a Standby network card, it will take over in the case of a problem with one of the primary network cards. This is only the case if the cards are the same card type (only Ethernet or FDDI).

Once the cluster interconnect is connected modify the following:

The /etc/cmcluster/cmclnodelist file is used by ServiceGuard OPS Edition for remote operations. If this file is not configured ServiceGuard OPS Edition will use the /.rhosts file.

Modify the /etc/cmcluster/cmclnodelist file:
Format of the file is:
[hostname] [rootuse] [#comment]

Example of a "/etc/cmcluster/cmclnodelist" equivalence file:

nodehp1 root #node1
nodehp2 root #node2
nodehp1 oracle # software owner on node1
hodehp2 oracle # software owneron node2

1.4 Unix User and Group Creation

Create a user who will own the Oracle RAC software. This must be done on each node in the cluster using the same uid and gid for each user.

1. Log in as the root user.

2. Create a unix database administrator group and oracle user:
Look at /etc/group, use any available gid this example uses 200

 # groupadd -g 200 dba

Look at /etc/passwd and ensure that the uid you select is not used. This example uses uid 300 and the newly created gid for the dba group.

 # useradd -u 300 -g 200 -d /home/oracle -s /usr/bin/csh -m oracle

---

2. Pre-Cluster Configuration

2.1 Modifying the /etc/lvmrc File as the root user

- Turn off automatic volume group activation to keep shared volume groups from being activated at boot time. You want the shared volume groups enabled with the cluster startup:

  Copy the /etc/lvmrc to a backup file
  # cp /etc/lvmrc /etc/lvmrc_orig:
  Modify the /etc/lvmrc:
  From:
  AUTO_VG_ACTIVATE=1
  To:
  AUTO_VG_ACTIVATE=0

- Add entries to the custom_vg_activation function in the /etc/lvmrc file for all volume groups that include logical volumes with non shared filesystems on them /etc/lvmrc example:

```
 custom_vg_activation()
{
# e.g. /sbin/vgchange -a y -s
# parallel_vg_sync "/dev/vg00 /dev/vg01"
# parallel_vg_sync "/dev/vg02 /dev/vg03"

/sbin/vgchange -a y vg00
/sbin/vgchange -a y vg01
/sbin/vgchange -a y vg02
/sbin/vgchange -a y vg03

return 0
}
```

2.2 Creating Shared Logical Volumes

- On the first node:
  pvcreate -f /dev/rdsk/c0t12d0 - Initialize direct access to a physical volume
  mkdir /dev/vg_ops - Create a directory in the /dev directory with the name of the volume group
  mknod /dev/vg_ops/group c 64 0x060000 - Major number will always be 64 but the minor number must be unique
  vgcreate /dev/vg_ops /dev/dsk/c0t12d0 - Create the volume group

vgchange -a y vg_ops - Activate the volume group

lvcreate -n P901_control_01.ctl -L 110 /dev/vg_ops - Create your logical volumes
lvcreate -n P901_control_02.ctl -L 110 /dev/vg_ops
lvcreate -n P901_control_03.ctl -L 110 /dev/vg_ops
lvcreate -n P901_system_01.dbf -L 400 /dev/vg_ops
lvcreate -n P901_log1_01.log -L 120 /dev/vg_ops
lvcreate -n P901_log1_02.log -L 120 /dev/vg_ops
lvcreate -n P901_log1_03.log -L 120 /dev/vg_ops
lvcreate -n P901_log2_01.log -L 120 /dev/vg_ops
lvcreate -n P901_log2_02.log -L 120 /dev/vg_ops
lvcreate -n P901_log2_03.log -L 120 /dev/vg_ops
lvcreate -n P901_spfile1.dbf -L 5 /dev/vg_ops
lvcreate -n P901_users_01.dbf -L 120 /dev/vg_ops
lvcreate -n P901_temp_01.dbf -L 100 /dev/vg_ops
lvcreate -n P901_undotbs_01.dbf -L 312 /dev/vg_ops
lvcreate -n P901_undotbs_02.dbf -L 312 /dev/vg_ops
lvcreate -n P901_example_01.dbf -L 160 /dev/vg_ops
lvcreate -n P901_cwmlite_01.dbf -L 100 /dev/vg_ops
lvcreate -n P901_indx_01.dbf -L 70 /dev/vg_ops
lvcreate -n P901_tools_01.dbf -L 20 /dev/vg_ops
lvcreate -n P901_drsys_01.dbf -L 90 /dev/vg_ops

vgchange -a n vg_ops - Deactivate the volume group
vgexport -v -s -p -m /tmp/vg_ops.map /dev/vg_ops - Create a map of the logical volume
rcp /tmp/vg_ops.map opcbhp2:/tmp/vg_ops.map - Copy the volume map to each node

- On the other nodes:
  mkdir /dev/vg_ops
  mknod /dev/vg_ops/group c 64 0x060000
  vgimport -v -s -m /tmp/vg_ops.map /dev/vg_ops - Create the volume group and logical volumes on the other nodes

- Modify the ownership an permissions on both nodes, this example shows the commands if being done from the node where the logical volumes were originally created:
  # chown oracle:dba /dev/vg_ops/r*
  # remsh nodehp2 chown oracle:dba /dev/vg_ops/r*
  # chmod 777 /dev/vg_ops
  # remsh nodehp2 chmod 777 /dev/vg_ops
  # chmod 660 /dev/vg_ops/r*
  # remsh nodehp2 chmod 660 /dev/vg_ops/r*

Note: When creating a new volume group the minor number must be unique when issuing the "mknod" command. You can use the following command to see which minor numbers are already in use:

 # find /dev -name group -exec ls -l {} \;

crw------- 1 root sys 64 0x060000 May 6 07:52 /dev/vg_ops/group

In this example the minor listed for "vg_ops" is "0x060000" so if creating a new volume group "0x060000" could not be used. Valid minor numbers range from "0x010000" to "0xFF0000".

2.3 Cluster Software Installation

The ServiceGuard OPS Edition is installed from the Applications CD-ROM of HP-UX using "**swinstall**"

2.4 Form a Cluster

1. Steps to configure the cluster:

Use cmquerycl to create a cluster configuration file

The cmquerycl command is used to create a default cluster configuration ascii file. You must specify each node that belongs to the cluster with the -n option. This needs to be performed on one node only.

 # cmquerycl -v -C /etc/cmcluster/cmclconf.ascii -n nodehp1 -n nodehp2

2. Modify the cluster configuration file

Parameters that needed modification on nodehp . Note that this cluster is running a minimal configuration. All other parameters in the ascii configuration file where set automatically such as heartbeat configuration.:

```
CLUSTER_NAME            #example nodehp
FIRST_CLUSTER_LOCK      # /dev/vg_ops
FIRST_CLUSTER_PV        # /dev/dsk/c0t12d0
OPS_VOLUME_GROUP        # /dev/vg_ops
DLM_ENABLED             # NO
GMS_ENABLED             # NO
```

Additonal Information:
FIRST_CLUSTER_LOCK, FIRST_CLUSTER_PV = The cluster lock volume group and disk are used for the same purpose as a quorum device. This volume group and disk must be accessible by all nodes. A failure of a node to access the lock volume will result with the node leaving the cluster.

OPS_VOLUME_GROUP = All cluster aware volume groups need to be identified
DLM_ENABLED and GMS_ENABLED = set to NO when using >8.1.7

3. Use cmcheckconf to verify the cluster configuration file

Verify the cluster configuration file
 # cmcheckconf -v -C /etc/cmcluster/cmclconf.ascii

4. Use vgchange to activate the lock volume group

Activate the lock volume group which is vg_ops on the nodehp cluster:
 # vgchange -a y vg_ops

5. Use cmapplyconf to copy the cluster configuration to all nodes

Use the cmapplyconf command to apply the configuration to all nodes:
 # cmapplyconf -v -C /etc/cmcluster/cmclconf.ascii

6. Use vgchange again to deactive the lock volume group

Deactivate the lock volume group:
 # vgchange -a n vg_ops

2.5 Basic Cluster Administration

Starting & Stopping Cluster Nodes

Starting the cluster:

cmruncl to start the cluster:
 # cmruncl

Use vgchange to mark shared volume groups shared from one node:
 # vgchange -S y -c y vg_ops

Use vgchange to activate shared logical volumes from each node:
 # vgchange -a s vg_ops
 # remsh nodehp2 /usr/sbin/vgchange -a s vg_ops

Use cmhaltcl to stop the cluster or cmhaltnode to remove a single node from the cluster:
 # cmhaltnode

Adding a Node Back Into a Running Cluster
Use cmrunnode to add the node back into a running cluster
 # cmrunnode

2.6 Log Files for Cluster

/var/adm/syslog/syslog.log contains the system log for ServiceGuard OPS Edition messages.
/var/adm/clic_log is used for HyperFabric interface log if this is being used.

3.0 Preparing for the installation of RAC

The Real Application Clusters installation process includes three major tasks.

1. Configure the shared disks and UNIX preinstallation tasks.
2. Run the Oracle Universal Installer to install the Oracle9i Enterprise Edition and the Oracle9i Real Application Clusters software.
3. Create and configure your database.

3.1 Configure the Shared Disks

Real Application Clusters requires that all each instance be able to access a set of unformatted devices on a shared disk subsystem. These shared disks are also referred to as raw devices. If your platform supports an Oracle-certified cluster file system, however, you can store the files that Real Application Clusters requires directly on the cluster file system.

The Oracle instances in Real Application Clusters write data onto the raw devices to update the control file, server parameter file, each datafile, and each redo log file. All instances in the cluster share these files.

The Oracle instances in the RAC configuration write information to raw devices defined for:

- The control file
- The spfile.ora
- Each datafile
- Each ONLINE redo log file
- Server Manager (SRVM) configuration information

It is therefore necessary to define raw devices for each of these categories of file. The Oracle Database Configuration Assistant (DBCA) will create a seed database expecting the following configuration:-

Raw Volume	File Size	Sample File Name
SYSTEM tablespace	400 Mb	*db_name*_raw_system_400m
USERS tablespace	120 Mb	*db_name*_raw_users_120m
TEMP tablespace	100 Mb	*db_name*_raw_temp_100m
UNDOTBS tablespace per instance	312 Mb	*db_name*_raw_undotbs*x*_312m
CWMLITE tablespace	100 Mb	*db_name*_raw_cwmlite_100m
EXAMPLE	160 Mb	*db_name*_raw_example_160m
OEMREPO	20 Mb	*db_name*_raw_oemrepo_20m
INDX tablespace	70 Mb	*db_name*_raw_indx_70m
TOOLS tablespace	12 Mb	*db_name*_raw_tools_12m
DRYSYS tablespace	90 Mb	*db_name*_raw_drsys_90m
First control file	110 Mb	*db_name*_raw_controlfile1_110m
Second control file	110 Mb	*db_name*_raw_controlfile2_110m
Two ONLINE redo log files per instance	120 Mb x 2	*db_name_thread_lognumber*_120m
spfile.ora	5 Mb	*db_name*_raw_spfile_5m
srvmconfig	100 Mb	*db_name*_raw_srvmconf_100m

Note: Automatic Undo Management requires an undo tablespace per instance therefore you would require a minimum of 2 tablespaces as described above. By following the naming convention described in the table above, raw partitions are identified with the database and the raw volume type (the data contained in the raw volume). Raw volume size is also identified using this method.

Note: In the sample names listed in the table, the string *db_name* should be replaced with the actual database name, *thread* is the thread number of the instance, and *lognumber* is the log number within a thread.

On the node from which you run the Oracle Universal Installer, create an ASCII file identifying the raw volume objects as shown above. The DBCA requires that these objects exist during installation and database creation. When creating the ASCII file content for the objects, name them using the format:

database_object=raw_device_file_path

When you create the ASCII file, separate the database objects from the paths with equals (=) signs as shown in the example below:-

```
system1=/dev/vx/rdsk/oracle_dg/db_name_raw_system_400m
spfile1=/dev/vx/rdsk/oracle_dg/db_name_raw_spfile_5m
users1=/dev/vx/rdsk/oracle_dg/db_name_raw_users_120m
temp1=/dev/vx/rdsk/oracle_dg/db_name_raw_temp_100m
undotbs1=/dev/vx/rdsk/oracle_dg/db_name_raw_undotbs1_312m
undotbs2=/dev/vx/rdsk/oracle_dg/db_name_raw_undotbs2_312m
example1=/dev/vx/rdsk/oracle_dg/db_name_raw_example_160m
cwmlite1=/dev/vx/rdsk/oracle_dg/db_name_raw_cwmlite_100m
indx1=/dev/vx/rdsk/oracle_dg/db_name_raw_indx_70m
tools1=/dev/vx/rdsk/oracle_dg/db_name_raw_tools_12m
drsys1=/dev/vx/rdsk/oracle_dg/db_name_raw_drsys_90m
control1=/dev/vx/rdsk/oracle_dg/db_name_raw_controlfile1_110m
control2=/dev/vx/rdsk/oracle_dg/db_name_raw_controlfile2_110m
redo1_1=/dev/vx/rdsk/oracle_dg/db_name_raw_log11_120m
redo1_2=/dev/vx/rdsk/oracle_dg/db_name_raw_log12_120m
redo2_1=/dev/vx/rdsk/oracle_dg/db_name_raw_log21_120m
redo2_2=/dev/vx/rdsk/oracle_dg/db_name_raw_log22_120m
```

You must specify that Oracle should use this file to determine the raw device volume names by setting the following environment variable where filename is the name of the ASCII file that contains the entries shown in the example above:

csh:

```
setenv DBCA_RAW_CONFIG filename
```

sh or ksh:

```
DBCA_RAW_CONFIG=filename; export DBCA_RAW_CONFIG
```

3.2 UNIX Preinstallation Tasks

After configuring the raw volumes, perform the following steps prior to installation as root user:

- Create a mount point directory on each node to serve as the top of your Oracle software directory structure so that:
  - The name of the mount point on each node is identical to that on the initial node
  - The `oracle` account has read, write, and execute privileges

- On the node from which you will run the Oracle Universal Installer, set up user equivalence by adding entries for all nodes in the cluster, including the local node, to the .rhosts file of the oracle account, or the `/etc/hosts.equiv` file.
- As oracle account user, check for user equivalence for the oracle account by performing a remote login (`rlogin`) to each node in the cluster.
- As oracle account user, if you are prompted for a password, you have not given the `oracle` account the same attributes on all nodes. You must correct this because the Oracle Universal Installer cannot use the `rcp` command to copy Oracle products to the remote node's directories without user equivalence.

Establish system environment variables

- Set a local **`bin`** directory in the user's PATH, such as `/usr/local/bin`, or `/opt/bin`. It is necessary to have execute permissions on this directory.
- Set the `DISPLAY` variable to point to the system's (from where you will run OUI) IP address, or name, X server, and screen.
- Set a temporary directory path for TMPDIR with at least 20 Mb of free space to which the

OUI has write permission.

Establish Oracle environment variables: Set the following Oracle environment variables:

Environment Variable	Suggested value
ORACLE_BASE	eg `/u01/app/oracle`
ORACLE_HOME	eg `/u01/app/oracle/product/901`
ORACLE_TERM	xterm
NLS_LANG	AMERICAN-AMERICA.UTF8 for example
ORA_NLS33	`$ORACLE_HOME/ocommon/nls/admin/data`
PATH	Should contain `$ORACLE_HOME/bin`

CLASSPATH	`$ORACLE_HOME/JRE:$ORACLE_HOME/jlib \`
	`$ORACLE_HOME/rdbms/jlib: \`
	`$ORACLE_HOME/network/jlib`

- Create the directory `/var/opt/oracle` and set ownership to the oracle user.

3.3 Using the Oracle Universal Installer for Real Application Clusters

Follow these procedures to use the Oracle Universal Installer to install the Oracle Enterprise Edition and the Real Application Clusters software. Oracle*9i* is supplied on multiple CD-ROM disks. During the installation process it is necessary to switch between the CD-ROMS. OUI will manage the switching between CDs. For the latest RAC/HP certification matrix see here.

To install the Oracle Software, perform the following:.

- Login as the oracle user
- **`$ ./<cdrom_mount_point>/runInstaller`**
- At the OUI Welcome screen, click **Next**.
- A prompt will appear for the Inventory Location (if this is the first time that OUI has been run on this system). This is the base directory into which OUI will install files. The Oracle Inventory definition can be found in the file **/var/opt/oracle/oraInst.loc**. Click **OK**.
- Verify the UNIX group name of the user who controls the installation of the Oracle*9i* software. If an instruction to run `/tmp/orainstRoot.sh` appears, the pre-installation steps were not completed successfully. Typically, the `/var/opt/oracle` directory does not exist or is not writeable by oracle. Run `/tmp/orainstRoot.sh` to correct this, forcing Oracle Inventory files, and others, to be written to the `ORACLE_HOME` directory. Once again this screen only appears the first time Oracle*9i* products are installed on the system. Click **Next**.
- The File Location window will appear. **Do NOT** change the Source field. The Destination field defaults to the `ORACLE_HOME` environment variable. Click **Next**.
- Select the Products to install. In this example, select the **Oracle*9i* Server** then click **Next**.
- Select the installation type. Choose the **Enterprise Edition** option. The selection on this screen refers to the installation operation, not the database configuration. The next screen allows for a customized database configuration to be chosen. Click **Next**.
- Select the configuration type. In this example you choose the Advanced Configuration as this option provides a database that you can customize, and configures the selected server products. Select **Customized** and click **Next**.
- Select the other nodes on to which the Oracle RDBMS software will be installed. It is not necessary to select the node on which the OUI is currently running. Click **Next**.
- Identify the raw partition in to which the Oracle*9i* Real Application Clusters (RAC) configuration information will be written. It is recommended that this raw partition is a minimum of 100MB in size.

- An option to Upgrade or Migrate an existing database is presented. Do **NOT** select the radio button. The Oracle Migration utility is not able to upgrade a RAC database, and will error if selected to do so.
- The Summary screen will be presented. Confirm that the RAC database software will be installed and then click **Install**. The OUI will install the Oracle*9i* software on to the local node, and then copy this information to the other nodes selected.
- Once Install is selected, the OUI will install the Oracle RAC software on to the local node, and then copy software to the other nodes selected earlier. This will take some time. During the installation process, the OUI does not display messages indicating that components are being installed on other nodes - I/O activity may be the only indication that the process is continuing.

3.4 Create a RAC Database using the Oracle Database Configuration Assistant

The Oracle Database Configuration Assistant (DBCA) will create a database for you (for an example of manual database creation see Database Creation in Oracle9*i* RAC). The DBCA creates your database using the optimal flexible architecture (OFA). This means the DBCA creates your database files, including the default server parameter file, using standard file naming and file placement practices. The primary phases of DBCA processing are:-

- Verify that you correctly configured the shared disks for each tablespace (for non-cluster file system platforms)
- Create the database
- Configure the Oracle network services
- Start the database instances and listeners

Oracle Corporation recommends that you use the DBCA to create your database. This is because the DBCA preconfigured databases optimize your environment to take advantage of Oracle*9i* features such as the server parameter file and automatic undo management. The DBCA also enables you to define arbitrary tablespaces as part of the database creation process. So even if you have datafile requirements that differ from those offered in one of the DBCA templates, use the DBCA. You can also execute user-specified scripts as part of the database creation process.

Note: Prior to running the DBCA it may be necessary to run the NETCA tool or to manually set up your network files. To run the NETCA tool execute the command netca from the $ORACLE_HOME/bin directory. This will configure the necessary listener names and protocol addresses, client naming methods, Net service names and Directory server usage. Also, it is recommended that the Global Services Daemon (GSD) is started on all nodes prior to running DBCA. To run the GSD execute the command gsd from the $ORACLE_HOME/bin directory.

The DBCA and the Oracle Net Configuration Assistant also accurately configure your Real Application Clusters environment for various Oracle high availability features and cluster administration tools.

- DBCA will launch as part of the installation process, but can be run manually by executing the command dbca from the `$ORACLE_HOME/bin` directory on UNIX platforms. The RAC Welcome Page displays. Choose **Oracle Cluster Database** option and select **Next**.
- The Operations page is displayed. Choose the option **Create a Database** and click **Next**.
- The Node Selection page appears. Select the nodes that you want to configure as part of the RAC database and click **Next**. If nodes are missing from the Node Selection then perform clusterware diagnostics by executing the `$ORACLE_HOME/bin/lsnodes -v` command and analyzing its output. Refer to your vendor's clusterware documentation if the output indicates that your clusterware is not properly installed. Resolve the problem and then restart the DBCA.
- The Database Templates page is displayed. The templates other than New Database include datafiles. Choose **New Database** and then click **Next**.

- The **Show Details** button provides information on the database template selected.
- DBCA now displays the Database Identification page. Enter the **Global Database Name** and **Oracle System Identifier (SID)**. The Global Database Name is typically of the form *name.domain*, for example ***mydb.us.oracle.com*** while the SID is used to uniquely identify an instance (DBCA should insert a suggested SID, equivalent to *name1* where *name* was entered in the Database Name field). In the RAC case the SID specified will be used as a prefix for the instance number. For example, ***MYDB***, would become ***MYDB1, MYDB2*** for instance 1 and 2 respectively.
- The Database Options page is displayed. Select the options you wish to configure and then choose **Next**. **Note**: If you did not choose New Database from the Database Template page, you will not see this screen.
- The Additional database Configurations button displays additional database features. Make sure both are checked and click **OK**.
- Select the connection options desired from the Database Connection Options page. **Note**: If you did not choose New Database from the Database Template page, you will not see this screen. Click **Next**.
- DBCA now displays the Initialization Parameters page. This page comprises a number of Tab fields. Modify the **Memory settings** if desired and then select the **File Locations** tab to update information on the Initialization Parameters filename and location. Then click **Next**.
- The option **Create persistent initialization parameter file** is selected by default. If you have a cluster file system, then enter a **file system name**, otherwise a **raw device name** for the location of the server parameter file (spfile) must be entered. Then click **Next**.
- The button **File Location Variables**… displays variable information. Click **OK**.
- The button **All Initialization Parameters**… displays the Initialization Parameters dialog box. This box presents values for all initialization parameters and indicates whether they are to be included in the spfile to be created through the check box, *included (Y/N)*. Instance specific parameters have an instance value in the instance column. Complete entries in the **All Initialization Parameters** page and select **Close**. **Note**: There are a few exceptions to what can be altered via this screen. Ensure all entries in the Initialization Parameters page are complete and select **Next**.
- DBCA now displays the **Database Storage** Window. This page allows you to enter file names for each tablespace in your database.
- The file names are displayed in the **Datafiles** folder, but are entered by selecting the **Tablespaces** icon, and then selecting the tablespace object from the expanded tree. Any names displayed here can be changed. A configuration file can be used, see section 3.1, (pointed to by the environment variable `DBCA_RAW_CONFIG`). Complete the database storage information and click **Next**.
- The **Database Creation Options** page is displayed. Ensure that the option **Create Database** is checked and click **Finish**.
- The **DBCA Summary** window is displayed. Review this information and then click **OK**.
- Once the Summary screen is closed using the OK option, DBCA begins to create the database according to the values specified.

A new database now exists. It can be accessed via Oracle SQL*PLUS or other applications designed to work with an Oracle RAC database.

4.0 Administering Real Application Clusters Instances

Oracle Corporation recommends that you use SRVCTL to administer your Real Application Clusters database environment. SRVCTL manages configuration information that is used by several Oracle tools. For example, Oracle Enterprise Manager and the Intelligent Agent use the configuration information that SRVCTL generates to discover and monitor nodes in your cluster. Before using SRVCTL, ensure that your Global Services Daemon (GSD) is running *after* you configure your

database. To use SRVCTL, you must have already created the configuration information for the database that you want to administer. You must have done this either by using the Oracle Database Configuration Assistant (DBCA), or by using the `srvctl add` command as described below.

If this is the first Oracle9*i* database created on this cluster, then you must initialize the clusterwide SRVM configuration. Firstly, create or edit the file `/var/opt/oracle/srvConfig.loc` file and add the entry `srvconfig_loc=`*`path_name`*.where the path name is a small cluster-shared raw volume eg

```
$ vi /var/opt/oracle/srvConfig.loc
srvconfig_loc=/dev/vx/rdsk/datadg/rac_srvconfig_10m
```

Then execute the following command to initialize this raw volume (Note: This cannot be run while the `gsd` is running. Before v9.2 you will need to kill the `.../jre/1.1.8/bin/...` process to stop the gsd from running):-

```
$ srvconfig -init
```

The first time you use the SRVCTL Utility to create the configuration, start the Global Services Daemon (GSD) on all nodes so that SRVCTL can access your cluster's configuration information. Then execute the `srvctl add` command so that Real Application Clusters knows what instances belong to your cluster using the following syntax:-

For Oracle RAC v9.0.1:-

$ gsd
Successfully started the daemon on the local node.

$ srvctl add db -p db_name -o oracle_home

Then for each instance enter the command:

$ srvctl add instance -p db_name -i sid -n node

To display the configuration details for, example, databases racdb1/2, on nodes racnode1/2 with instances racinst1/2 run:-

$ srvctl config
racdb1
racdb2

$ srvctl config -p racdb1
racnode1 racinst1
racnode2 racinst2

$ srvctl config -p racdb1 -n racnode1
racnode1 racinst1

Examples of starting and stopping RAC follow:-

$ srvctl start -p racdb1
Instance successfully started on node: racnode2
Listeners successfully started on node: racnode2
Instance successfully started on node: racnode1

Listeners successfully started on node: racnode1

$ srvctl stop -p racdb2
Instance successfully stopped on node: racnode2
Instance successfully stopped on node: racnode1
Listener successfully stopped on node: racnode2
Listener successfully stopped on node: racnode1

$ srvctl stop -p racdb1 -i racinst2 -s inst
Instance successfully stopped on node: racnode2

$ srvctl stop -p racdb1 -s inst
PRKO-2035 : Instance is already stopped on node: racnode2
Instance successfully stopped on node: racnode1

For Oracle RAC v9.2.0+:-

$ gsdctl start
Successfully started the daemon on the local node.

$ srvctl add database -d db_name -o oracle_home [-m domain_name] [-s spfile]

Then for each instance enter the command:

$ srvctl add instance -d db_name -i sid -n node

To display the configuration details for, example, databases racdb1/2, on nodes racnode1/2 with instances racinst1/2 run:-

$ srvctl config
racdb1
racdb2

$ srvctl config -p racdb1 -n racnode1
racnode1 racinst1 /u01/app/oracle/product/9.2.0.1

$ srvctl status database -d racdb1
Instance racinst1 is running on node racnode1
Instance racinst2 is running on node racnode2

Examples of starting and stopping RAC follow:-

$ srvctl start database -d racdb2

$ srvctl stop database -d racdb2

$ srvctl stop instance -d racdb1 -i racinst2

$ srvctl start instance -d racdb1 -i racinst2

$ gsdctl stat
GSD is running on local node

$ gsdctl stop

For further information on srvctl and gsdctl see the Oracle9i Real Application Clusters Administration manual.

5.0 References

- Note 154360.1 - HP: Quick Start Guide - 9.0.x RDBMS Installation
- Note:189256.1 - Script to Verify Installation Requirements for Oracle 9.x version of RDBMS
- Note:137288.1 - Database Creation in Oracle9*i* RAC
- RAC/HP certification matrix
- Oracle9i Real Application Clusters on HP-UX
- Oracle*9i* Real Application Clusters Installation and Configuration Release 1 (9.0.1)
- Oracle*9i* Real Application Clusters Concepts
- Oracle*9i* Real Application Clusters Administration
- Oracle*9i* Real Application Clusters Deployment and Performance
- Oracle*9i* Installation Guide for Compaq Tru64, Hewlett-Packard HPUX, IBM-AIX, Linux, and Sun Solaris-based systems.
- Oracle*9i* Release Notes

.