

Oracle Homes in an Oracle Real Application Clusters Environment

An Oracle White Paper
January 2008

Table Of Contents

Introduction	3
Software	3
Oracle Central Inventory	5
Oracle Home Inventory	6
Rolling Upgrades	7
Cloning Oracle Software	7
Non-shared Oracle Home	9
Advantages	9
Disadvantages	9
More about the Disadvantages.....	10
Shared Oracle Home	11
Shared Oracle Base	12
Binaries in a Shared Home.....	12
Configuration Files in Shared Home.....	12
Log Files in Shared Home	12
Oracle Clusterware in a Shared Home.....	13
OCFS2 as a Shared Oracle Home	14
NFS as a Shared Oracle Home	14
Advantages	15
Disadvantages	15
More about the Disadvantages.....	15
Maintaining Shared and Non-Shared Oracle Homes	16
Using Enterprise Manager's Provisioning Automation.....	16
Installing Software.....	16
Silent Installations	16
Upgrading Software	17
Applying One-Off Patches	17
Applying Patchsets	17
Adding Nodes.....	18
Removing Nodes.....	18
Update Node List.....	19
Instance Placement	19
Conclusion.....	20
APPENDIX A: ORACLE Clusterware upgrade output	21
APPENDIX B: References	22
Oracle RAC 11g:.....	22
Oracle RAC 10g Release 2:	22

Oracle Homes in an Oracle Real Application Clusters Environment

“To share or not to share”, that is the question.

INTRODUCTION

It is obvious that each node in a clustered environment needs the Oracle software in order to access an Oracle Real Application Clusters (RAC) database. What is not so obvious is whether the software should be installed in a shared fashion (shared Oracle Home) or privately on each node. This paper will discuss the pros and cons of each method as well as explain how to complete common operations, such as add/remove nodes, using OPatch and Oracle’s Universal Installer (OUI). The complete manual steps will be described however it should be noted that all of the described steps are completely automated with Enterprise Manager’s Life Cycle Management (Provisioning Automation)¹ features. The discussion applies to Oracle RAC 10g and Oracle RAC 11g. Further reading material is provided in the reference section.

SOFTWARE

Traditionally in a clustered environment, the Oracle software is installed privately on each server that needs access to it. However, let’s consider a 16-node cluster, installing, managing, patching and backing up can become a bit complex with a lot of copies of the same software. This is where a shared Oracle Home can help, as the software is installed only once, on a shared file system. This single software and configuration location may reduce the complexity of managing and configuring the software in a clustered environment. Further in the discussion we’ll discuss the pros and cons of using each approach in depth.

The following are some shared file systems available today:

Clustered file systems are available on all major platforms.

Table 1 - Clustered file systems on major operating systems.

Cluster file system	Provider	Platforms
CFS (Cluster File System)	HP Corporation	Tru64 (part of TruCluster software)
GPFS (General Parallel	IBM Corporation	AIX, Linux & POWER

¹ http://www.oracle.com/technology/products/oem/mgmt_solutions/provisioning.html has plenty of screen-watches, self-running demos, data sheets, etc.

File System)		
NFS (Network File system)	Certified NFS device	Supported NFS Server/Client (Most OSs)
OCFS2 (Oracle Cluster File System 2)	Oracle Corporation	Linux
OCFS (Oracle Cluster File System)	Oracle Corporation	Windows (part of Oracle RAC software)
Matrix Server	HP (Polyserve)	Linux, Windows
GFS (Global File System)	Red Hat	Red Hat Enterprise Linux
StorEdge QFS	Sun Microsystems	Solaris
Veritas CFS (Cluster File System) VxCFS	Symantec	Solaris, HP-UX, AIX, Linux (Storage Foundation for Windows also available)

Table 1 Notes:

- Third party file systems are not directly certified by Oracle, the partners help in this process, see the Certify section on Metalink for official support policy. For Linux the [RAC Technologies Matrix](#) on OTN is a useful starting point.
- The above list is known to be correct at the time of publication and does not intend to list all available options.
- The certification process of a clustered file system is different for shared database files support, and shared Oracle Home support. For example on Linux, OCFS1 only supports database files but **not** shared Oracle Home; for shared home support use OCFS2.

The discussion of shared vs. non-shared Oracle Home is very similar for the Database and Automatic Storage Management (ASM); any specific differences will be mentioned explicitly. In this paper, we assume that the Database and ASM are installed in separate Oracle Homes as recommended by Oracle. Although it is possible to install the Oracle Clusterware in a shared Oracle Home, **it is not recommended**. The Clusterware Home consists of system level software and as such should be treated as a special case and installed privately on each node as it contains sensitive software necessary for the health of the node. It is possible to mix shared and non-shared homes for each type of Oracle Home, for example, during testing, transitioning from one setup to another, you can setup a non-shared Clusterware Home with shared-ASM/Database homes or any other combination that makes sense for the given environment.

The following are some definition of terms that will be useful for the discussion:

Although technically it is possible to install the Oracle Clusterware in a shared home fashion, it is not recommended.

Oracle Central Inventory

The Oracle Central Inventory holds information on all software installed as well as cluster members. Each node has a private copy, even when a shared home is used.

Regardless of whether you choose a shared or non-shared Oracle Home, each node must still have its own *private* Oracle software Central Inventory. Unix systems support multiple Central Inventory locations whereas Windows supports only one location. If \$ORACLE_BASE is set, the Central Inventory default location in Oracle Database 10g is set to: \$ORACLE_BASE/oraInventory and in Oracle Database 11g to: \$ORACLE_BASE/./oraInventory, otherwise \$HOME/oraInventory is used. In a silent installation, described further in the document, the Central Inventory location is specified by the INVENTORY_LOCATION variable. The Central Inventory may be placed anywhere the DBA desires, and the actual value is stored in the oraInst.loc² text file in the inventory_loc variable. For easy navigation to the Central Inventory, a copy of this file is located at the top of each Oracle Home. On Windows the registry key HKLM\SOFTWARE\Oracle\inst_loc points to the Central Inventory location. The Oracle Central Inventory is a small repository that holds information about the Oracle software that is installed on that node only; it **cannot** be placed on a clustered/shared filesystem to be shared by other nodes in an Oracle RAC environment. In fact, even OUI sessions from the same node cannot be run concurrently due to the locking of the Central Inventory for the duration of an installation. If such an attempt is made the following errors will be displayed:

```
SEVERE: oracle.sysman.oii.oiit.OiitTargetLockNotAvailableException:
OUI-10004: The target area /opt/orabase/oraInventory is being
written to by another session.
Please close all the other (write) sessions for accessing
inventory.
```

The Inventory lock file is <Central Inventory>/locks/writer.lock, which is private for each node. In single instance installations it is possible to have more than one Central Inventory per node if more concurrency is needed, possibly across several DBA groups³, however in a RAC environment this should be avoided. This is because the Clusterware must exist in each Central Inventory, attaching it later can be done but adds unneeded complexity, hence in RAC maintain one Central Inventory per node.

Below is a sample Central Inventory file

<inventory_loc>/oraInventory/ContentsXML/inventory.xml (viewable using a text editor), we can see a Clusterware Oracle Home and the nodes that are associated with that Oracle Home, also known as, the “cluster members”:

```
<HOME NAME="crs10g" LOC="/crs10g" TYPE="O" IDX="1" CRS="true">
  <NODE_LIST>
    <NODE NAME="node1"/> <NODE NAME="node2"/>
    ...
    <NODE NAME="node15"/> <NODE NAME="node16"/>
  </NODE_LIST>
</HOME>
```

² On Solaris, HP-UX, Tru64 in /var/opt/oracle; on Linux and AIX in /etc

³ Technically, each DBA group could have a different Central Inventory hence multiple OUI sessions can run concurrently, each against a different Central Inventory, essentially managing different software homes on that node (should be avoided in RAC).

Such an association of nodes exists for each installed Oracle Home. The cluster members stored in the Oracle Home inventory (next section) take precedence over these values in the Central Inventory, which in this context is used as a fallback mechanism. This information is maintained and consulted by Oracle Universal Installer (OUI) during installations as well as by OPatch when applying a patch, unless the `-local` flag is used (`-local` says to apply patch only to current node, not to all nodes associated with the Oracle Home). This flag will be discussed later in this paper. Do not manually edit the inventory XML (Extensible Markup Language) files as OUI provides supported methods to manipulate these files. See the Reference section of the OUI documentation or Maintenance section of this paper for detailed examples.

Oracle Home Inventory

As seen above, the Oracle Central Inventory that is privately kept on each node points to all the installed Oracle Homes on that node. In *each* Oracle Home in the “inventory” subdirectory lives the Oracle Home Inventory. The information in the Oracle Home inventory is also consulted by OUI and OPatch and overrides the one found in the Central Inventory. The inventory files hold very useful information, however they should not be edited or modified manually. For example:

```
$ORACLE_HOME/inventory/ContentsXML/comps.xml
```

Holds all the products, versions, patchsets, interim patches installed on this Oracle Home. The DBA should use `opatch lsinventory` to view the contents of this file. Another example is the file:

```
$ORACLE_HOME/inventory/ContentsXML/oraclehomeproperties.xml
```

Here is a sample of the contents:

```
<ORACLEHOME_INFO>
  <GUID>182700754#.1304811780</GUID>
  <HOME/>
  <ARU_PLATFORM_INFO>
    <ARU_ID>46</ARU_ID>
    <ARU_ID_DESCRIPTION>Linux x86</ARU_ID_DESCRIPTION>
  </ARU_PLATFORM_INFO>
  <CLUSTER_INFO>
    <NODE_LIST>
      <NODE_NAME="node1"/> <NODE_NAME="node2"/>
      ...
      <NODE_NAME="node15"/> <NODE_NAME="node16"/>
    </NODE_LIST>
  </CLUSTER_INFO>
</ORACLEHOME_INFO>
```

We can see the platform id (46 Linux x86) and the 16-nodes associated with this Oracle Home. There may be rare cases where the two inventories (central and per-home) get out of sync on a node basis or across cluster members. We will show how to check and update the two inventories to be in sync in a supported fashion. The above sample output does not contain the `LOCAL_NODE` tag, which means that this is a shared Oracle Home. In a private Oracle Home, in addition to the list of

cluster members, there will be a LOCAL_NODE tag with the local node name; this is used by OUI as the node name until it can consult the Clusterware to verify it.

Rolling Upgrades

A particularly interesting method of updating the Oracle software in a RAC environment is referred to as Rolling Upgrade. Using this method⁴ a patch is applied in sequence on each Oracle Home using OPatch (for one-off patches) or OUI (in the case of patchsets) thus eliminating cluster wide database downtime. Complete details on Rolling Upgrades can be found in the standard Oracle documentation⁵, however here are a few points to keep in mind:

Applying patches in a rolling upgrade fashion can completely eliminate cluster wide database downtime.

- It is not possible to do rolling upgrades when a shared Oracle Home is used. This is because there is a single copy of the software shared by all nodes.
- It is always possible to do a rolling upgrade of the Oracle Clusterware provided a non-shared Oracle Home is used.⁶
- Starting with Oracle RAC 11g, ASM can be upgraded in a rolling upgrade fashion⁷, including patchsets.
- Without a standby database, RDBMS (10g & 11g) or ASM (10g) rolling upgrades are only supported (leveraging multiple instances in RAC) for some one-off patches and not for patchsets or full major releases. To determine if a one-off patch can be applied in a rolling upgrade fashion see it's README or issue one of the following commands:

Unix/Linux: `$ opatch query [patch location] | grep rolling`

Or:

Windows: `C:\> opatch query [patch location] | findstr rolling`

For completeness, if a standby database is available then rolling upgrades are possible for patchsets as well as full releases (Oracle Database 10g or Oracle Database 11g) with minimal downtime even if RAC is not installed. See Metalink [Note 300479.1](#) and SQL Apply Rolling Upgrade Best Practices paper on OTN for more details.

Cloning Oracle Software

Often times, to cut down on installation and deployment time, the Oracle software is simply copied from one location to another on local or remote nodes. This is supported by following a process called cloning and is fully described in Chapter 6 of the OUI and OPatch User's Guide 11g. The first step is as mentioned to simply copy the Oracle Home (ASM, Clusterware or Database) from one location to another

Cloning can speed up deployment time.

⁴ Shutdown instance, apply patch, startup instance, continue to next node.

⁵ The Oracle Database 11g Upgrade Guide, Chapter 1.

⁶ Refer to Oracle Clusterware Installation Guide 11g (Appendix A) or Metalink [Note: 338706.1](#) for details on Oracle Clusterware rolling upgrades.

⁷ See Oracle Database 11g Storage Admin Guide regarding ASM rolling upgrades.

(using tar, zip, cp, etc.), preserving the ownership and permissions, and excluding log files and other host specific directories. The next step is to run the Perl script `<Oracle Home>/clone/bin/clone.pl` on the destination, which does the following operations:

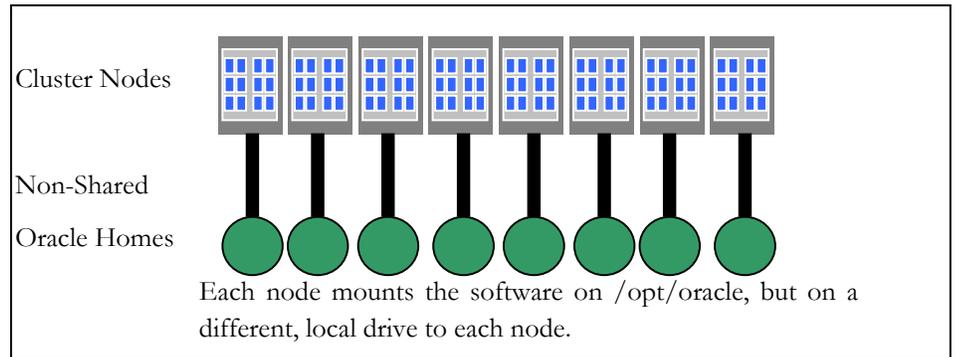
1. Parse all the files under the new Oracle Home and modifies the text in them to be the new home path.
2. Attach the new Oracle Home to the Central Inventory and correct all that is needed in the new Oracle Home specific inventory files.
3. Re-link all binaries in the new Oracle Home.

After the script is completed the cloned Oracle Home is fully functional as if it was freshly installed, for operations such as patching or upgrading using OPatch or OUI. Skipping the `clone.pl` script is not supported as the copied home still has references to the old Oracle Home paths and it is not attached to the inventories properly.

NON-SHARED ORACLE HOME

In a non-shared Oracle Home, sometimes referred to as Private Oracle Home, each node in the cluster maintains a complete copy of the Oracle software⁸ tree on local storage, as depicted in figure 2:

Figure 2: Non-Shared Oracle Home



Although non-shared Homes take additional disk space and time during install, the benefits typically are well worth it.

This is the most common way in which Oracle Real Application Clusters are installed. Although it takes additional disk space and time to make these copies during install, the benefits typically are well worth it. For example, it is very useful in the case of applying patches, as each node can be dealt with separately, thus allowing service to the database to continue even during a patching operation (rolling upgrade). All the Oracle tools fully support this configuration and make it easy to manage.

Non-shared Oracle Home, in summary:

Advantages

- Ability to apply certain one-off patches in a rolling upgrade fashion. Only the patched instance is brought down for the duration of the patch application (using OPatch), hence service to the database is never interrupted. Not all patches are allowed to be applied in this fashion. See Rolling Upgrade section earlier for more details.
- Software tree is not a single point of failure; any damage to it will only affect a single node.

Disadvantages

- More time consuming having to manage or apply patches to each node. Always have to make sure that all patches have been applied to all nodes, using `opatch lsinventory` to confirm all nodes were patched

Non-shared Oracle Home is recommended in most cases due to the abilities to patch in a rolling upgrade fashion with zero downtime.

⁸ In this context, Oracle software refers to Oracle RAC (Database), Oracle ASM or Oracle Clusterware.

correctly⁹. In many cases, a checksum utility such as `md5sum` may be used to compare binary checksum, however this is unreliable since some linkers embed the link timestamp in the binary and that may be the source of the mismatched checksum¹⁰.

- Configuration and Log files are spread across all nodes, making it harder to view them in a central place.

More about the Disadvantages

Although it's true that managing multiple separate Oracle Homes can be more time consuming, there are several tools that can help reduce that complexity.

For example, OPatch is cluster-aware and by default applies a patch to all nodes associated with that Oracle Home as stored in the Oracle inventories (to apply to a single node use the `-local` flag).

Using cluster-aware tools such as *pdsh* can greatly assist in managing clustered environments.

There are other external tools that can help with distributed operations; for example, on the Linux platform *pdsh* (Parallel Distributed Shell <http://www.llnl.gov/linux/pdsh/>) is a useful utility that allows seamless remote execution of commands in a clustered environment. A configuration file (`/usr/local/etc/machines`) holds all members of the cluster thus allowing *pdsh* to execute a single command cluster-wide or on any subset of nodes.

As for accessing log files and trace file directories in a central way, it is easy to setup a shared file system to hold the traces for all cluster members. In Oracle Database 10g set both `user_dump_dest` and `background_dump_dest` to the shared filesystem location. In Oracle Database 11g relocating the Automatic Diagnostic Repository (ADR) is simply done by issuing:

```
$ alter system set diagnostic_dest='/ocfs2' scope=both;
```

A “diag” directory (with further subdirectories) will automatically be created under `/ocfs2` and all tracing will go there immediately.

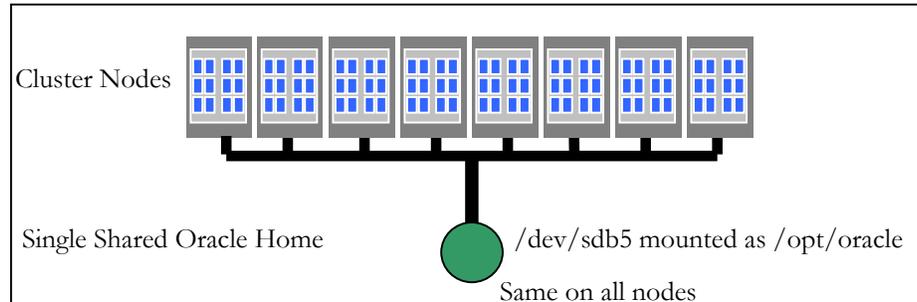
⁹ The latest version of OPatch adds a patch verification phase to ensure all files were copied correctly, see Metalink [Note: 353150.1](#) for details.

¹⁰ See Metalink [Note: 443460.1](#) for an example of different checksums.

SHARED ORACLE HOME

In a shared Oracle Home, all nodes in the cluster use the same physical copy of the software, as depicted in figure 1:

Figure 1: Shared Oracle Home



Implementing a shared Oracle Home installation is *straightforward*; simply supply a shared mount point at the OUI (Oracle Universal Installer) interview stage for the Oracle Home location.

This simplifies configuration and management of many database operations, as there's a single central location rather than on each node. Implementing a shared Oracle Home installation is *straightforward*; simply supply a shared mount point at the Oracle Universal Installer (OUI) interview stage for the Oracle Home location. In the example below `/opt/oracle` will be the mount point on *all* nodes in the cluster:

```
# mount -t ocfs2 /dev/sdb5 /opt/oracle
```

This location must be a supported clustered file system, such as a certified NFS server or OCFS2.¹¹

The installer will automatically detect that this is a shared location and will bypass the software copy/propagation stage to remote nodes, i.e. install binaries only once on the shared file system, thus cutting down install time, especially for larger clusters. During a shared Oracle Home installation in the install log you will see:

```
INFO: FASTCOPYRAC: Oracle Home < /opt/oracle/db-10201> is shared.
Instantiating RAC files
...
INFO: This is a shared oracle home or remote nodes are null.
No copy required.
```

Instead of the typical non-shared home entry:

```
INFO: FASTCOPYRAC: Oracle Home < /opt/oracle/db-10201> is not
shared. Instantiating RAC files
...
INFO: Updating files in Oracle home '/opt/oracle/db-10201' to
remote nodes 'node2,node3,node4,...,node16'.
```

Software files will only be propagated to remote nodes in the non-shared case. In the shared home case there is no need to copy the software, as it is shared across all nodes using the shared clustered file system. You may use CVU to check if a location is shared between nodes as follows:

¹¹ On Linux, OCFS1 only supports shared data file placement, not the software tree, so it can **not** be used to host a shared Oracle home; OCFS2 is fully RAC certified for both Software and Data; see **Table 1** above for more details.

```
$ cluvfy comp cfs -f /opt/oracle/db-10201 -n node1,node2,...,node16
```

Shared Oracle Base

It is possible to also share the Oracle Base directory, however care should be taken on the placement of the Oracle Central Inventory, since in some cases OUI may pick its location to be relative to \$ORACLE_BASE. If during initial installation you choose to place \$ORACLE_BASE on a shared file system, this could possibly lead to all nodes accessing the same Central Inventory. As stated above, the Oracle Central Inventory must be private to each node, and cannot be written to from more than one node. Therefore, if you choose to place the Oracle Base on a shared location, make sure the Central Inventory location is explicitly defined (during initial installation) to a local file system available on each node.

Binaries in a Shared Home

In most cases executable binaries have local dependencies, for example, to a specific JAVA version or OS shared libraries. If these binaries are shared across all nodes, it limits the ability to upgrade or modify these nodes without impacting the ability to run these shared binaries without artifacts. For example: An OS upgrade may replace a dependent library that requires the shared binary to be re-linked; if such a re-link is performed, that shared binary will break on other nodes which do not yet have the upgraded library. For this reason care should be taken to ensure OS compatibility when using a shared Oracle Home.

Configuration Files in Shared Home

All Oracle configuration files have a cluster-aware syntax or separate directory location to enable multi-node support. For example, the database initialization parameters in the `init.ora` file have the syntax of:

```
[*.|SID.]<parameter>=<value>
```

which means that each parameter can be set database-wide or specific for each instance. Hence we can have a single `init.ora` file, or the recommended `spfile`, supporting all nodes. A similar mechanism exists in configuration files; for example, in the `listener.ora` configuration file the node name is part of the default listener name:

```
LISTENER_NODE1 = (DESCRIPTION_LIST = ... HOST=node1 ...  
LISTENER_NODE2 = (DESCRIPTION_LIST = ... HOST=node2 ...  
LISTENER_NODE3 = (DESCRIPTION_LIST = ... HOST=node3 ...
```

thus allowing all listeners to be defined in a single `listener.ora` file.

Log Files in Shared Home

In Oracle Database 11g, the Automatic Diagnostic Repository (ADR) consolidates all tracing and logging to one central location designated by the `diagnostic_dest` `init.ora` parameter. It fully supports both shared and non-shared locations as each node has a separate subdirectory.

By default in Oracle Database 10g, all the Oracle directories, log files and trace files already have the SID and node names or numbers (under Oracle Clusterware Homes) embedded in them which means no additional setup is required to make the Oracle software tree RAC aware. Minor exceptions are the files under the `audit` and `cdump` directories, which do not have the SID pre-pended to the filename. It is therefore advised to relocate them using the initialization parameters: `audit_file_dest` and `core_dump_dest` respectively, such that each node has a different location to avoid accidental overwrites from multiple processes using the same filename from different nodes.

In Oracle Database 10g, by default the database logs and trace files reside under:

```
$ORACLE_BASE/admin/<dbname>
```

If the `$ORACLE_BASE` was not selected to be on shared file system then each node will write to that local file system. In this case you can setup a shared log directory by issuing:

```
SQL> alter system set background_dump_dest='<shared-location>'12
scope=both;
```

Using a central shared log/trace repository, DBAs only have to look in one location for logs or traces from all cluster members making it a much easier task managing the cluster.

Having a central shared log/trace repository means that DBAs only have to look in one location for logs or traces from all cluster members rather than login to each server to view the logs or traces making it a much easier task managing the cluster. As explained earlier, it is also possible to configure a central logging directory in a non-shared Oracle Home, if a shared file system exists it is possible to share just log/trace files and not the software.

To balance the above statements, depending on the configuration (hardware, OS, selected clustered file system) it is possible that during a major outage on large number of nodes, many instances will attempt to dump diagnostic traces all at the same time, which may put a heavy load on the clustered file system and causing the tracing operation to take much longer than needed. In those cases, it might be beneficial to offload these tracing activities to a local file system using `init.ora` parameters or symbolic links from the clustered file system to the local file system. It was observed in certain cases that such heavy write activity could cause cluster availability issues, however it highly depends on the hardware and software used.

Oracle Clusterware in a Shared Home

As mentioned earlier, the Oracle Clusterware software should not be placed on a shared filesystem. Two reasons for this are:

1. If all cluster members use the *same copy* of the software, application of a patch using OPatch or a patchset using OUI in a rolling upgrade fashion is not possible. This is also clearly documented in all the Clusterware patch's READMEs. A cluster-wide outage must be scheduled.

¹² `<shared-location>` would be replaced with the directory name on the shared file system. Similar steps for `user_dump_dest`.

2. It is a *single* point of failure (SPOF). For example, if a user accidentally deletes some binaries under the Oracle Clusterware Shared Home, then the Clusterware on **all** nodes will be affected immediately, potentially causing a cluster-wide outage or severe malfunction due to the missing binaries. Another example is the popular NFS-mounted shared Oracle Home or shared Clusterware Home. Consider the case where the network cable is pulled. Now the Clusterware cannot function since all its software and logic is on the NFS filesystem, which is not available, processes hang (in many cases even a `kill -9` is not effective in clearing them), and cluster resources go into UNKNOWN state, plus other undesirable conditions.

OCFS2 as a Shared Oracle Home

If you use OCFS2 as a shared Oracle Home, carefully review [OCFS2's FAQ section](#) (on <http://oss.oracle.com/>). Specifically notice that OCFS2¹³ does not currently support writeable mmap files which are used by the EM agent¹⁴ and ASM¹⁵. You **must** relocate them to a filesystem that supports writeable mmap files (e.g. ext3); it need not be shared across all nodes. Shutdown the agent and issue:

```
$ mv $ORACLE_HOME/dbs/hc_<SID>.dat /<mmap_filesystem>/hc_<SID>.dat
$ ln -s /<mmap_filesystem>/hc_<SID>.dat $ORACLE_HOME/dbs/hc_<SID>.dat
```

Follow the same procedure for the ASM file as well.

Additionally, the OCFS2 timeouts should carefully be set according to the FAQ and Metalink Notes [434255.1](#) & [395878.1](#) to avoid node reboots by the OCFS2 cluster monitor.

NFS as a Shared Oracle Home

In Oracle RAC 11g, the VKTM and LMS processes are configured to run in real-time priority (SCHED_RR) using the `$ORACLE_HOME/bin/oradism` binary running as `setuid root`¹⁶. Typically, for security reasons NFS mounts are configured to prevent root `setuid` executions on exported filesystems. If the Oracle Home is shared on such an NFS filesystem, follow the workarounds documented in the release notes to either reconfigure the NFS server to allow `setuid root` execution or relocate this binary to a local filesystem (must be same path on all nodes) or any other filesystem that supports `setuid root` (e.g. OCFS2) as follows:

On all nodes, copy and ensure correct permissions and ownership, as root:

```
# cp $ORACLE_HOME/bin/oradism /<setuid_filesystem>/oradism
# chown root.oinstall /<setuid_filesystem>/oradism
# chmod 4750 /<setuid_filesystem>/oradism
```

When completed, issue these commands from any one node:

```
$ rm $ORACLE_HOME/bin/oradism
$ ln -s /<setuid_filesystem>/oradism $ORACLE_HOME/bin/oradism
```

¹³ OCFS2 1.4 release is expected to support writeable mmap files.

¹⁴ EM Agent health check file is: `$ORACLE_HOME/dbs/hc_<ORACLE_SID>.dat`

¹⁵ ASM file is `$ASM_HOME/dbs/ab_<ORACLE_SID>.dat`

¹⁶ Reference Metalink [Note: 461144.1](#) for additional such binaries.

To confirm the processes' run priority (sample commands on Linux):

```
$ ps -eo pid,class,rtprio,command | grep -e vktm -e lms
6092 RR      1 ora_vktm_orcl1
6114 RR      1 ora_lms0_orcl1
6118 RR      1 ora_lms1_orcl1
```

Notice the “RR” which stands for SCHED_RR and “rtprio” field is 1, signifying the processes are correctly set to run in real-time priority. If neither workaround is followed the priority will be time-sharing or “TS” and the “rtprio” field null.

Shared Oracle Home, in summary:

Advantages

- Faster installation, as software is not propagated to all nodes.
- Easier access to log files and traces from all nodes to all cluster members.
- Easier to maintain and less error prone for the user; files always ‘in sync’.

Disadvantages

- Inability to apply patches (using OPatch) or re-link the software to enable/disable features, in a rolling upgrade fashion, since all nodes use the same software tree.
- Requires cross-node OS compatibility due to local dependency of the shared binaries (See section “Binaries in a Shared Home” earlier in this section)
- Introduces a single point of failure; if the software tree is damaged in any way, the entire cluster will be affected.

More about the Disadvantages

The single shared home is in fact a SPOF (Single Point Of Failure). However, in some cases, a simple restore of the filesystem would get the system back online. Generally the software tree changes infrequently, making it very easy to backup and quickly restore in case of filesystem failures. It is recommended to protect the software tree with some RAID technology, but note that RAID will not prevent human errors or file system related corruptions. A well-defined backup strategy and RAID protection for the shared Oracle Home may not prevent an outage, but may minimize it sufficiently such that service level targets can be maintained. Some sites may not be impacted severely by taking down the entire cluster for a cluster wide upgrade.

Shared Oracle Home results in faster installation of the software as well as an easier to maintain environment. However, it introduces a single point of failure and prevents rolling upgrades.

MAINTAINING SHARED AND NON-SHARED ORACLE HOMES

Follow the standard methods as already documented in standard documentation as well as patch/set READMEs. Here are some common operations and ways to accomplish them in a shared or non-shared Oracle Homes.

Using Enterprise Manager's Provisioning Automation

As briefly mentioned in the introduction, Enterprise Manager's (EM) Provisioning Automation elegantly automates all the steps described in this section, by the use of Deployment Procedures¹⁷. These are reusable, extensible and hot-pluggable building blocks that allow full automation of operations, such as adding or removing nodes and patching. The clear advantage is that the entire software stack is taken into account, and extended. If the Clusterware is missing it will be installed, same for ASM, and other components. The patching engine is connected online to Metalink allowing automated patch delivery and application. EM fully supports shared or non-shared Oracle Homes. Full details can be found at the links provided in the introduction and this section.

Enterprise Manager's Provisioning Automation elegantly automates scale-up and scale-down operations and much more.

Installing Software

Follow the standard methods of installing Oracle software using OUI; it fully supports shared and non-shared Oracle Homes for the Clusterware, Database or ASM. Remember that the Oracle Clusterware version must always be the highest, for full support matrix of the three types of homes (Clusterware, ASM, RDBMS), see Metalink [Note: 337737.1](#)

Silent Installations

Typically Oracle Universal Installer (OUI) is invoked in GUI (Graphical User Interface) mode, however you may wish to automate installation of the Oracle software or may not have a graphical terminal available. An extremely easy technique to get a quick start with silent installs is by invoking OUI with the `-record` flag to produce a response file from the current installation session, which could later be used for a silent installation (no GUI pop up for user input or error reporting), using the `-silent` and `-responseFile` flags. Note that for security reasons, the passwords are not recorded in the response file so you will have to edit the text file and manually add them so that the silent install will not fail due to missing passwords (in response file look for: `s_superAdminSamePasswd & s_superAdminSamePasswdAgain`). If you change the installation source you may need to adjust the `FROM_LOCATION` variable, typically set to `../stage/products.xml`, it should point to the `products.xml` file on the installation media relative to the `oraparams.ini` file that lives in the `install` subdirectory, hence the `..`. Another useful OUI flag is the `-force`, which allows silent mode installations into a non-empty directory, for example `lost+found`

In all cases, silent installation is a great way to speed up and automate Oracle installations.

¹⁷ Using Deployment Procedures to Scale-Up or Scale-Down:
<http://www.oracle.com/technology/products/oem/pdf/grid-automation-deployment-procedures.pdf>

subdirectory or some application specific files that must reside on the top level for some reason. It should not be used to recover from failed installations; proper cleaning of the destination folder should be followed in such cases. The progress report is printed to standard output and as in any standard OUI invocation, install action logs are kept in both the <Central Inventory>/logs subdirectory and in \$ORACLE_HOME/db/cfgtoollogs/oui directory. Log files from remote nodes are copied to the local node with the remote hostname suffixed; all files from one invocation use the same (local node) timestamp for easy correlation.

Upgrading Software

In any type of software upgrade you should use the same type of Oracle Home for the upgrade, you should not change from a non-shared Oracle Home to a shared Oracle Home during an upgrade.

Applying One-Off Patches

Follow the standard methods of applying patches using the OPatch utility. Make sure you download the latest OPatch version from Metalink (10.2 bug placeholder: 4898608; 10.1 bug placeholder: 2617419). In the shared home case it is **required** to add the `-local` flag so the patch gets applied correctly, only from one node, since it is a shared Oracle Home, all nodes will get the fix. In the shared Oracle Home case, it is not possible to apply the one-off patch in a rolling upgrade fashion; rather, a cluster wide database outage should be scheduled when OPatch is invoked. If applying a Clusterware fix and the Oracle Clusterware version is different than the Oracle Database or ASM versions special care should be taken, please review Metalink [Note: 363254.1](#) for details.

Applying Patchsets

Follow the standard methods of applying a patchset using OUI. Refer to patchset README for details. In summary, the general steps are:

- First, apply the patchset to the Oracle Clusterware Home on all nodes.¹⁸ If using non-shared Clusterware Home then this can be done in a rolling upgrade fashion. With shared Oracle Clusterware Oracle Home you must shutdown the entire cluster to apply a patchset.
- Next, apply the patchset to the Database Oracle Home on all nodes. Similar considerations should be followed for an ASM home, with the exception that starting with Oracle ASM 11g, rolling upgrades are fully supported. It is not possible to apply the Database portion of a patchset (10.2.0.3 for example on top of a 10.2.0.2) in a rolling upgrade manner, unless a logical standby database is in place. Therefore, a full cluster wide database outage should be scheduled to apply the patchset. In the non-shared Oracle Home case it is possible to reduce cluster-wide downtime:

¹⁸ See Appendix A for a sample Clusterware rolling upgrade output.

- Shutdown all instances of all databases (on this node) using the to-be-patched Oracle Home.
- Apply the Database portion of the patchset on that local node only, using the `-local` flag when invoking OUI (Instances on remote nodes are not affected and are still running).
- After patchset application completes on above chosen node, shutdown *all* remaining instances of all databases using that Oracle Home on all remote nodes.
- Startup (exclusive) the database using the newly patched Oracle Home and complete the post-patchset steps (running catalog scripts against database, etc., as per patchset README).
- Concurrently, apply patchset to all remaining nodes invoking OUI with the `-enableRollingUpgrade` flag. In the cluster nodes page select all nodes except the single already-patched node.
- When catalog scripts complete, bring up instances on all remaining nodes. The duration of the catalog scripts run is not be dependent on the number of nodes in the cluster.

Adding Nodes

In both shared and non-shared Oracle Home, follow the standard methods of adding a node to the cluster as documented in the standard documentation (Oracle RAC 11g Administration and Deployment Guide)¹⁹. The installer will detect if the software tree needs to be copied, in the non-shared Home case, and will do it automatically. In summary the general steps are:

- Invoke `oui/bin/addNode.sh` from a Clusterware Home of an existing cluster member, this will pop up the GUI OUI interface and allow you to add the node(s) to the cluster and, if needed, will copy the Clusterware software to the remote nodes(s).
- Next, invoke `oui/bin/addNode.sh` from a Database Oracle Home of an existing cluster member, this will pop up the GUI OUI interface which will complete the Database portion of the node(s) addition, and, if needed, will copy the Database software to the remote nodes(s).

Removing Nodes

In both shared and non-shared Oracle Home, follows the standard methods of removing a node from the cluster as documented in Oracle RAC 11g Administration and Deployment Guide²⁰. The note contains instructions for shared homes; basically you don't want to accidentally delete the shared software upon node removal. For

¹⁹ Oracle RAC 10g Release 1 (10.1.x) refer to Metalink [Note: 270512.1](#).

²⁰ Oracle RAC 10g Release 1 (10.1.x) refer to Metalink [Note: 269320.1](#).

example, in a shared home case you would “detach” the Oracle Home and in a non-shared home you can safely uninstall it.

Update Node List

In certain cases the cluster node list maintained in the Oracle Central Inventory and Oracle Home Inventory files may get out of sync, for example a software install took place when a node was unavailable or a restore of an Oracle Home was done from an earlier point in time²¹. You can visually inspect the inventory files mentioned at the beginning of this document to determine if such out-of-sync occurred²². The node lists can be updated by using the `-updateNodeList` flag of Oracle Universal Installer as follows (do not edit the Oracle Inventory files manually):

Do not edit the Oracle Inventory XML files manually; use the appropriate OUI commands.

```
$ORACLE_HOME/oui/bin/<runInstaller or setup.exe> -updateNodeList  
-silent ORACLE_HOME="<oracle home path>"  
  "CLUSTER_NODES={node1,node2,node3}"
```

The above command will automatically run on the three specified nodes and set the cluster members of the specified Oracle Home in the Central Inventory and the Oracle Home inventory to the three specified nodes: node1, node2 and node3. If using a shared Oracle Home, you *must* also specify the `-cfs` flag to indicate this is a shared Oracle Home. On a shared Oracle Home, another option is to invoke OUI with the `-local` flag, which will execute the update node list command only on the local node. However, since the Central Inventory on remote nodes will not be updated when using the `-local` flag, this command will have to be executed on all nodes to achieve the desired effect.

Make sure you see the success message from the update node list operation as:

```
'UpdateNodeList' was successful.
```

Otherwise check the OUI log files under the Oracle Home, located in:

```
$ORACLE_HOME/db/cfgtoollogs/oui
```

Instance Placement

In Oracle Database 10g, the Oracle Clusterware is responsible for automatic startup and shutdown of instances (as well as other services or user defined applications) in the cluster using a predefined association of an instance to a node. This association is stored in the OCR file and can easily be changed at any time using the `srvctl` utility; if for example node12 is down for maintenance you may assign its instance to node4:

```
$ srvctl modify instance -d racdb -i racdb12 -n node4
```

The above command will assign instance racdb12 to node4 instead of its default node, probably node12. The change will take affect the next time the Oracle Clusterware shuts down and starts up this instance. Of course the needed files

²¹ For full list of inventory operations and recovery procedures refer to: Oracle® Universal Installer and OPatch User’s Guide 11g Release 1 (11.1), section 2.6.

²² Incorrect node list may include symptoms such as software install only to nodes specified in that list rather than to all cluster members.

(init.ora, password file, etc.) for the racdb12 instance need to exist on the target node, in this case node4.

Instance placement can also be overridden manually; any instance could be manually started from any node/Oracle Home as long as the configuration files, specifically the init.ora file, are kept updated to support that (initracdb12.ora available on node4's Oracle Home in above example). For the case of shared Oracle Home this is done automatically since all files reside in one location. In a non-shared Oracle Home, the DBA will have to symbolically link or copy these files in order to facilitate optional instance placement. For example, in a non-shared Oracle Home, DBCA will create a single init.ora private to each Oracle Home, the DBA should copy that file to remote Oracle Homes if they plan on starting up that instance from these nodes. The password file `orapw<ORACLE_SID>` should also be copied, since it might be needed to startup the database using SQL*Plus.

CONCLUSION

Managing the Oracle software in a RAC environment is easy whether you choose a shared or non-shared Oracle Home. In most cases, the non-shared Oracle Home is the preferred solution due to the ability to patch in a rolling upgrade fashion with zero downtime, the elimination of the single point of failure, and binary dependency issues. Oracle Enterprise Manager Life Cycle Management (Provisioning Automation) features offer significant ease of use with *full* automation of all manual operations described in this paper.

**Non-shared Oracle Home:
Recommended in most cases!**

**Shared Oracle Home: Supported, but
adds single point of failure and
harder rolling upgrades.**

APPENDIX A: ORACLE CLUSTERWARE UPGRADE OUTPUT

Sample output from an Oracle Clusterware rolling upgrade, 10.2.0.1 to 10.2.0.4:

Before upgrade:

```
[node1]$ crsctl query crs softwareversion
CRS software version on node [node1] is [10.2.0.1.0]
[node1]$ crsctl query crs activeversion
CRS software version on node [node1] is [10.2.0.1.0]
```

Notice how both versions are the same in the above output.

After only node1 has been upgraded to 10.2.0.4 (rest of nodes still run 10.2.0.1):

```
[node1]$ crsctl query crs softwareversion
CRS software version on node [node1] is [10.2.0.4.0]
[node1]$ crsctl query crs activeversion
CRS software version on node [node1] is [10.2.0.1.0]
```

Notice how the software version is higher than the active clusterware version.

After all nodes have been upgraded to 10.2.0.4:

```
[node1]$ crsctl query crs softwareversion
CRS software version on node [node1] is [10.2.0.4.0]
[node1]$ crsctl query crs activeversion
CRS software version on node [node1] is [10.2.0.4.0]
```

Now both versions are the same and all nodes have been upgraded to the newer version of the clusterware.

When the last node to be upgraded starts up the Oracle Clusterware stack with the new version, in the clusterware logs (crsd.log) we see:

```
<timestamp>: [CRSD][3086870208]0Daemon Version: 10.2.0.4.0 Active
Version: 10.2.0.1.0
<timestamp>: [CRSD][3086870208]0Active Version is less than
Software Version
<timestamp>: [CRSD][90975152]0New Active Version:10.2.0.4.0
<timestamp>: [CRSD][90975152]0Active Version changed to 10.2.0.4.0
```

Notice how the Oracle Clusterware maintains the lower version as the “active” protocol for the clusterware during the upgrade. Once all nodes have the newer version installed, the stack automatically switches to the newer version thus completing the upgrade process.

APPENDIX B: REFERENCES

The following official Oracle documents have references to shared and non-shared Oracle Home or are otherwise relevant, as follows:

Oracle RAC 11g:

[B28263-02](#) Oracle® Clusterware Installation Guide 11g Release 1 (11.1) for Linux

Installing (or rolling upgrade) the Oracle Clusterware 11g software.

[B28264-02](#) Oracle® Real Application Clusters Installation Guide 11g Release 1 (11.1) for Linux and UNIX

Installing the 11g RAC software.

[B28255-02](#) Oracle® Clusterware Administration and Deployment Guide 11g Release 1 (11.1) for Linux

Cloning / Removing the Oracle Clusterware software on shared and non-shared Oracle Homes.

[B28254-02](#) Oracle® Real Application Clusters Administration and Deployment Guide 11g Release 1 (11.1)

Adding or removing nodes in shared or non-shared Oracle Homes.

[B28300-02](#) Oracle® Database Upgrade Guide 11g Release 1 (11.1)

Explains rolling upgrades in an Oracle RAC environment.

[B31207-01](#) Oracle® Universal Installer and OPatch User's Guide 11g Release 1 (11.1) for Windows and UNIX

General usage of Oracle Universal Installer as well as OPatch in shared Oracle Home is explained, for example, the -local flags for OPatch and OUI. This is the most extensive shared Oracle Home documentation available.

Oracle RAC 10g Release 2:

[B14203-08](#) Oracle® Database Oracle Clusterware and Oracle RAC Installation Guide 10g Release 2 (10.2) for Linux

Basic explanation of shared Oracle Home.

[B14197-04](#) Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide 10g Release 2 (10.2)

Detailed explanation of managing shared Oracle Home, adding or removing nodes, with exact commands in each case, shared and non-shared home.

[B14238-01](#) Oracle® Database Upgrade Guide 10g Release 2 (10.2)

Explains rolling upgrades in an Oracle RAC environment.

[B16227-04](#) Oracle® Universal Installer and OPatch User’s Guide 10g Release 2 (10.2) for Windows and UNIX

Using Oracle Universal Installer as well as OPatch in shared Oracle Home is explained, for example, the –local flags for OPatch.

[Metalink Note: 338706.1](#) Cluster Ready Services (CRS) rolling upgrade

[Metalink Note: 300479.1](#) Rolling Upgrades with Logical Standby

[Metalink Note: 302806.1](#) IBM General Parallel File System (GPFS) and Oracle RAC on AIX 5L and IBM eServer pSeries

[OTN](#): “SQL Apply Rolling Upgrade Best Practices”

HIGHLY RECOMMENDED



[OTN](#): “Best Practices for Optimizing Availability During Planned Maintenance Using Oracle Clusterware and Oracle Real Application Clusters”

Enterprise Manager Life Cycle Management – Provisioning Automation:

http://www.oracle.com/technology/products/oem/mgmt_solutions/provisioning.html has plenty of screen-watches, self-running demos, data sheets, etc.

Using Deployment Procedures to Scale-Up or Scale-Down:

<http://www.oracle.com/technology/products/oem/pdf/grid-automation-deployment-procedures.pdf>



Oracle Homes in an Oracle Real Application Clusters Environment

January 2008

Author: Saar Maoz

Contributing Authors: Roland Knapp, John McHugh, Philip Newlan, Juan Telez, Jonathan Creighton, Bharat Paliwal, Kirk McGowan, Barb Lundhild, Dan Norris

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

Worldwide Inquiries:

Phone: +1.650.506.7000

Fax: +1.650.506.7200

oracle.com

Copyright © 2008, Oracle. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice.

This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission. Oracle, JD Edwards, and PeopleSoft, are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.