

An Oracle White Paper
March 2011

Configuration of SAP NetWeaver for Oracle Grid Infrastructure 11.2.0.2 and Oracle Real Application Clusters 11g Release 2: A Best Practices Guide

Preface.....	5
Basic Setup: Prepare the SAP System.....	6
Option 1: New SAP System Install and Upgrade to Oracle 11.2.0.2	6
Option 2: Upgrade Single Instance Database to Oracle 11.2.0.2.....	7
Option 3: Upgrade RAC enabled Database to Oracle RAC 11.2.0.2	8
System Preparation for Oracle GRID Infrastructure before Installation.	10
Check MTU Size on Network Cards.....	10
Request DNS Subdomain Entry for the Cluster.....	10
Prepare Users and Groups for the Oracle Grid Installation.....	11
Install Oracle Grid Infrastructure Software on your Cluster	14
Network Option 1: Use Oracle GNS Service for Name Resolution....	15
Network Option 2: Use existing DNS without GNS.....	17
All Installations: Specify Network Interfaces.....	19
Storage Option 1: Place OCR and Voting Files on Oracle ASM.....	19
Storage Option 2: Shared Filesystem for OCR and Voting Files.....	20
All installations: ORACLE_BASE for GRID Infrastructure.....	21
Oracle Clusterware Parameter Adjustments for SAP Installations.....	25
Install Oracle RAC 11.2.0.2 Software on a Cluster.....	27

Prepare the Database Upgrade.....	32
Pin all cluster nodes in Oracle Clusterware.....	32
Re-install Oracle 10.2 Database Software and Patchsets.....	33
Change the ownership of database files.....	35
Check /etc/oratab.....	36
Upgrade the Oracle Database using DBUA.....	36
Modify Location for Server Parameter File spfile<SID>.ora.....	37
Manual Upgrade of the Database.....	39
After the Upgrade: Manual or by DBUA.....	39
Apply all required Patches for SAP	40
Database Control File Parameters.....	41
Automatic Undo Management with Undo Tablespaces.....	43
Redo Log Groups and Members.....	45
Modify Oracle Initialization Parameters for RAC.....	50
Oracle Password File.....	53
Enable the Cluster Database and Start all Instances.....	54
Oracle Network Configuration.....	55
Listener Configuration File for Node and SCAN Listeners.....	55
Network Configuration File tnsnames.ora.....	57

Option 1: Use EZCONNECT for Clients and Database Instances.....	58
Option 2: Use tnsnames.ora for Client Connections	58
Copy the tnsnames.ora File to the Location given in TNS_ADMIN...	62
Service Registration in CRS for the RAC enabled SAP Database.....	63
Configure the SAP Database and the RAC Instances in OCR	63
Define Database Services for SAP R/3 Application Instances	64
SAP User Environment.....	67
SAP Instance Profile	67
SAP START Profile	69
SAP J2EE Server Configuration for RAC Database	70
Define Database Services for J2EE Instances	71
SAP BR*Tools.....	73
Implementation and Configuration of SAP Enqueue Replication.....	73

Preface

This document explains all the necessary steps to configure an SAP system for Oracle Real Application Clusters 11g Release 2. The purpose of this paper is to give a detailed overview of all the necessary steps to upgrade an already existing Oracle RAC 10.2 system or perform a migration from a single-instance Oracle 11.2.0.2 database to a RAC database. It is intended to act as a template for use in your specific system environments. It will give you information on configuring your SAP R/3 system accordingly. If your system is already Oracle 10.2 RAC enabled, you will find the changes required to use new functionality from Oracle 11g. Differences with respect to single-instance migration will be pointed out.

This documentation is only for upgrading Oracle systems running on **UNIX** and **Linux** operating systems. The Windows operating system is covered in a separate document.

This white paper does not discuss the specific setup of the cluster hardware, the operating system and the storage subsystem used for your cluster solution. All pre-installation steps necessary for setting up the underlying cluster hardware are considered as minimum requirements. Please refer to the documentation of the respective platform and operating system vendor for details on how to set up the cluster hardware. Additional and up-to-date information about supported configurations are documented in SAP Support Note 527843. It is strongly recommended that you check the latest updates for your operating system, storage solution, etc. This paper does however cover those aspects that are necessary for using Oracle Grid Infrastructure; the Oracle Clusterware that ships with Oracle RAC 11.2.0.2 and that is used in conjunction with the Oracle database software. This paper describes the supported RAC configurations for an SAP installation that have been successfully tested.

Important note: This guide helps you to set up a RAC-enabled SAP system. It describes all required changes to the Oracle database, Oracle network configuration, Oracle instance parameters etc. But it is not complete: You must follow all additional steps addressed in SAP Installation and Upgrade Guides for Oracle Database 11g Release 2 as well. Please also refer to all RAC-related SAP notes. Start checking for Oracle RAC related information from SAP OSS note 527843.

Basic Setup: Prepare the SAP System

Preparations need to be made depending on the kind of SAP system you want to enable for Oracle RAC 11.2.0.2.

It is only supported to migrate to Oracle RAC 11.2.0.2 from one of the following configurations:

- New system install with Single Instance Oracle 10.2.0.4 or later and upgrade to Single Instance Oracle 11.2.0.2
- Migrate Single Instance 11.2.0.2 database to Oracle RAC 11.2.0.2
- Upgrade to Oracle RAC 11.2.0.2 from Oracle RAC 10.2.0.4 or later

The tasks that must be performed before the actual RAC configuration can start vary accordingly. Any major deviations between the systems are pointed out throughout the document.

Option 1: New SAP System Install and Upgrade to Oracle 11.2.0.2

Please refer to the Install and Configuration Guides from SAP to perform a first system installation. This installation of your SAP system will create a single instance Oracle database. There is no option available from SAP to install an Oracle RAC enabled database directly. The migration to Oracle RAC must be performed after the initial installation of the SAP system is completed.

Before the SAP installation actually starts, the hardware setup of the cluster should have finished. Place the database files, redo log files, archive logs and control files on a shared filesystem. By doing so, there is no need to copy the data to a shared storage location for RAC enabling later on.

This document assumes that the planned new Oracle RAC installation uses a shared filesystem (cluster filesystem or NAS NFS) for the database files, control files, redo log files, archived redo log files and executables under \$ORACLE_HOME (software installation directory for the Oracle RDBMS software), as well as all SAP executables in the /sapmnt directory. Throughout the rest of this document we will use the term shared filesystem for a

cluster filesystem or an NAS NFS filesystem. Other shared filesystem types than these two are not supported.

Installing the Oracle software in a shared filesystem is classified as an installation with a "Shared Oracle Home".

Note: SAP with RAC requires a Shared Oracle Home. Oracle Grid Infrastructure software is installed locally on each node of the cluster.

The voting disks and the OCR repository files with a new Oracle RAC 11.2.0.2 installation can be placed on a shared filesystem or on Oracle ASM. Raw devices for storing the voting disks and the OCR repository files are not supported.

During installation of the SAP software, the users <sid>adm and ora<sid> will be created. With a shared filesystem, the home directory for these users can be shared among all nodes in the cluster. This is not a mandatory requirement, but all configuration tasks with respect to user profiles etc. can be completed faster as the changes are valid for all the nodes in the cluster immediately. Administrative overhead by doing all changes multiple times is reduced. All examples given in this white paper assume a shared user home for users <sid>adm and ora<sid>.

Note: Probably you must install your SAP system with Oracle Database Release 10.2.0.4 or later as SAPInst cannot install all products versions of the SAP system with Oracle Database Release 11.2.0.2. After the initial installation, perform a single instance upgrade to Oracle Database Release 11g Release 2 before starting the migration to Oracle Real Application Clusters 11g Release 2.

Option 2: Upgrade Single Instance Database to Oracle 11.2.0.2

If you are running a single instance Oracle Database 10g Release 2 or older, you must perform an upgrade to single instance Oracle Database Release 11.2.0.2 before using this documentation. All necessary RAC migration steps described in this paper assume that all SAP installations are using at least a single instance Oracle Database 11.2.0.2.

The most recent updates on SAP Install and Upgrade documentation for the Oracle database can be found on the SAP Service Marketplace website, <http://service.sap.com/instguides>. Navigate to → *Other Documentation* → *Database Upgrades* → *Oracle* for the required guides.

Note: You cannot upgrade a 10.2 single instance database directly to a RAC 11.2.0.2 database using DBUA. If you start with a single instance Oracle 10gR2 database it is required to upgrade the database to a single instance Oracle 11.2.0.2 database first by following the procedures described by SAP and perform the RAC enabling of the database in a second step.

After finishing the single instance database upgrade to Oracle 11.2.0.2, follow the steps required for a migration to Oracle RAC.

You must move all Oracle database files, control files, redo log files, archived redo log files to a shared storage location on a shared filesystem before you start the RAC migration process. Put all SAP subdirectories (sapbackup, sapcheck, sapreorg, saptrace, oraarch etc.) in a shared filesystem location as well.

For easier maintenance of user profiles and settings, we recommend to move the home-directories for SAP users ora<SID> and <SID>adm to a shared filesystem.

Option 3: Upgrade RAC enabled Database to Oracle RAC 11.2.0.2

If the SAP system is already a cluster installation running with Oracle RAC 10.2.0.4 or later, then this document also addresses all the necessary upgrade steps. If you are just planning to upgrade the database software to Oracle Database 11g Release 2, most of the configuration changes in the database are already in place. With a Oracle RAC 10.2 enabled database for SAP, UNDO tablespaces or redo threads for the database instances etc. are already configured. However, you still need to change the Oracle network configuration in order to be able to use the simplified connection setup for connections to database services for SAP application instances. Whenever you can skip configuration steps, this is pointed out.

Note that if the number of nodes or database instances changes, you have to adjust the database configuration accordingly.

Either way, please also refer to the Upgrade Guides from SAP to check for minimum requirements.

Note: If your Oracle 10.2 database is already RAC enabled, you can use DBUA to perform the database upgrade step. If you plan to use DBUA, you must reinstall Oracle 10.2 software and patchsets as user oracle:oinstall.

This document assumes that the planned installation uses a shared filesystem for the datafiles, control files, redo log files, archived redo log files and executables under

\$ORACLE_HOME (software installation directory for the Oracle RDBMS software), as well as all SAP executables in the /sapmnt directory.

Note: With Oracle 10.2 RAC, Oracle Clusterware software and all related files have to reside on a shared filesystem. Oracle Grid Infrastructure 11g Release 2 software is stored on local filesystems on each node of the cluster.

Before you start the installation of Oracle 11.2.0.2 Grid software, shut down your existing Oracle RAC 10.2 system. Stop all running CRS daemons. Edit the file /etc/inittab and delete the following lines

```
h1:35:respawn:/etc/init.d/init.evmd run >/dev/null 2>&1 </dev/null
h1:35:respawn:/etc/init.d/init.cssd fatal >/dev/null 2>&1 </dev/null
h1:35:respawn:/etc/init.d/init.crsd run >/dev/null 2>&1 </dev/null
```

Execute “init –q” after saving the file. After that, check that no CRS processes are running. If so, you can either kill the processes or reboot the system.

Note to SAPCTL users: You must install a new version of SAPCTL designed for Oracle Clusterware 11.2.0.2. The SAPCTL version you run with Oracle CRS 10.2. is not compatible with the Oracle 11.2.0.2 cluster framework.

System Preparation for Oracle GRID Infrastructure before Installation

Check MTU Size on Network Cards

The MTU size on all network cards used for cluster communication must be of identical size. It is recommended for installation purposes that all network cards use an MTU size of 1500 bytes. Identify the network cards used for the public network and the cluster interconnect. Set MTU size to 1500 bytes for all these network cards.

Note: The installation will fail if MTU size is not identical on all network cards.

Request DNS Subdomain Entry for the Cluster

An important new feature of Oracle Grid Infrastructure 11.2.0.2 software is the ability to access cluster resources like database services by a SCAN (Single Client Access Name) address. The SCAN address is the only address clients need to know to establish a connection to the cluster. SCAN naming simplifies all configuration tasks on the client side. By using an SCAN address it is possible to add or remove cluster nodes without the need to change client configurations, e.g entries in tnsnames.ora for SAP application servers or the connect descriptor in the SAP Secure Store for the SAP JAVA stack.

The Oracle Grid Infrastructure 11.2.0.2 software for a cluster distributes SCAN listeners among available cluster nodes. To provide easy administration without the burden to maintain configuration files if cluster nodes are added or removed, Oracle Grid software provides a GNS daemon, which interacts with the DNS server and DHCP server in your organization.

If you **do not want** to use GNS, request an entry in your DNS server which resolves three IP addresses. These IP addresses must be on the same subnet as your public network.

Example:

```
cluster1-scan.my-company.com    IN A 10.165.110.166
                                IN A 10.165.110.167
                                IN A 10.165.110.168
```

If you **want** to use GNS, request a DNS subdomain for your cluster. Choose a meaningful subdomain name for your installation, like the name for the cluster you give during the grid infrastructure software installation process.

The GNS nameserver entry must resolve to a valid IP address in the public network of the cluster. Request this to be added by your network admin to the DNS configuration.

Example:

```
$ORIGIN cluster1.my-company.com.  
@                IN NS gns.cluster1.my-company.com.  
gns.cluster1.my-company.com IN A 10.165.111.103
```

Note: We recommend the use of GNS and avoid manual setup of cluster naming.

For other example on setting up DNS and DHCP on Linux, see MOS note 946452.1

Prepare Users and Groups for the Oracle Grid Installation

Oracle Grid Infrastructure software should be installed under the default Oracle user account which will be the owner of the software. We recommend the username oracle for this user. This is the default owner for all Oracle software in Oracle Best Practices recommendations.

Note: Starting with Oracle Grid Infrastructure Release 11.2.0.1, we recommend to follow the established Oracle standards for users and groups. Please note that with Oracle Release 10g the old rules for user/group membership are still in place. The new installation schema is only valid starting with Oracle Grid Infrastructure Release 11.2.0.1.

Note: With Oracle Grid Infrastructure components Oracle Clusterware and Oracle ASM, a fine grained permission model is available which we strongly recommend to use in combination with your SAP installation.

To leverage this model, you need to create the following groups on every node in the cluster. Make sure that the group numbers match on all nodes. You must create these groups only if they do not already exist from previous installations.

```
# groupadd -g 600 oinstall  
# groupadd -g 601 dba  
# groupadd -g 602 oper
```

The following groups are not required if you upgrade an existing Oracle RAC 10.2 database with no change in the location of the OCR and voting files on a shared filesystem. Create these groups only if you plan to place OCR and voting files in Oracle ASM:

```
# groupadd -g 603 asmadmin
# groupadd -g 604 asmoper
# groupadd -g 605 asmdba
```

Create or modify the following user on every node in the cluster:

```
# useradd -u 400 -g oinstall -G dba,asmdba,asmadmin -d /usr/home oracle
```

The user oracle has the primary group oinstall and is the software owner of all Oracle software starting with Oracle RAC 11.2.0.1 databases. User oracle also belongs to group dba. If you want to manage the Oracle ASM instances and the Oracle ASM storage by user oracle as some sort of superadmin, you can assign the groups asmdba, asmoper and asmadmin to user oracle. If your security constraints require separate users for storage administration tasks you must create additional users and assign them to the appropriate groups asmdba, asmoper and asmadmin.

Note: The home directory of user oracle must be local on every node in the cluster. You should not use a shared or remote filesystem like NFS to store the home directory as during the boot process this user account must be available for start of several daemon processes.

Log on to the first node as user oracle. Create directory ~/.ssh, if it does not exist already. Go to this directory.

```
$ mkdir .ssh
$ cd .ssh
```

Create private and public key files for RSA and DSA authorization:

```
$ ssh-keygen -t rsa -f <path_to_homedirectory>/.ssh/id_rsa <nodename>
```

Do not specify a passphrase, just hit Enter when asked.

```
$ ssh-keygen -t dsa -f <path_to_homedirectory>/.ssh/id_dsa_<nodename>
```

Once again, do not specify a passphrase. Hit Enter.

Repeat this on all nodes. Remember that you must log on to every node, as key generation is dependent on the node itself. After all private and public key files have been generated; you must create a config file containing pointers to the identity file for every node.

These identity files contain private keys for the respective nodes. You must specify the full pathname to these files. Example for the file ~/.ssh/config:

```
ForwardX11 no
PasswordAuthentication no
Host oracx1
IdentityFile /saphome1/oracle/.ssh/id_rsa_oracx1
IdentityFile /saphome1/oracle/.ssh/id_dsa_oracx1
Host oracx2
IdentityFile /saphome1/oracle/.ssh/id_rsa_oracx2
IdentityFile /saphome1/oracle/.ssh/id_dsa_oracx2
Host oracw1
IdentityFile /saphome1/oracle/.ssh/id_rsa_oracw1
IdentityFile /saphome1/oracle/.ssh/id_dsa_oracw1
Host oracw2
IdentityFile /saphome1/oracle/.ssh/id_rsa_oracw2
IdentityFile /saphome1/oracle/.ssh/id_dsa_oracw2
```

Copy all files containing public keys from all nodes to a temporary directory on one node. Then concatenate these files containing public keys from all nodes to the file `authorized_keys`.

```
$ cat *.pub > authorized_keys
```

Copy the file `authorized_keys` to the `.ssh` directory in the home directory of user `oracle` and change the file permissions as shown below. Execute this on every node.

```
$chmod 600 config
$chmod 600 authorized_keys
$cd
$chmod 700 .ssh
$chmod 755 ~
```

You should now be able to open a ssh connection to all hosts without being prompted for a password. Try this by executing

```
$ssh <nodename> date
```

Repeat this on all nodes in order to have the fingerprints recorded in the file `known_hosts`.

Check the Oracle documentation for all other minimum requirements before starting the installation. Oracle provides a tool for verifying that the conditions for a successful installation have been met. This tool is called `cluvfy` and is shipped with the Oracle software.

Note: Use the tool `cluvfy` to verify that all minimum requirements are met.

Note: Before you start the installation, shut down an existing Oracle 10.2 cluster. Stop all CRS processes currently running and edit the file `/etc/inittab`, comment out the line with startup directive. Execute an `init -q` command, or reboot the node.

Install Oracle Grid Infrastructure Software on your Cluster

This task must be completed before the Oracle RAC database software is installed. Oracle Grid Infrastructure software is the combination of Oracle Clusterware (CRS) and Oracle Automatic Storage Management (Oracle ASM) software. As both components must be installed into the same directory, we will refer to this directory as the Grid installation directory `GRID_HOME` throughout this paper.

Note: The 11.2 Grid Infrastructure software must be installed locally on each node of the cluster. This is an important change to Oracle Clusterware 10g installations with SAP.

Log on as user `oracle` and start the installation with the Oracle Universal Installer

Installation Option	Install and Configure Grid Infrastructure Upgrading an existing installation will require many additional changes which are not documented in this whitepaper. Only the installation option “Install and Configure Grid Infrastructure for a Cluster” is supported by SAP.
Installation Type	Advanced Installation
Product Language	English

Network Option 1: Use Oracle GNS Service for Name Resolution

We recommend that you make use of the Grid Naming Service (GNS) for simplified configuration of the Oracle network (tnsnames.ora and listener.ora). Specify a name for the cluster. The SCAN name depends on the network subdomain given for the cluster and optional on the cluster name.

The GNS subdomain is the name of the network subdomain you need to obtain from your network administrator. The GNS VIP address is the IP address to which DNS name resolution requests will be forwarded.

All IP addresses for node VIPs, SCAN VIPs and other services requiring an IP address will automatically be requested from a DHCP server in your network.

If you want to use GNS, request a DNS subdomain for your cluster. Choose a meaningful subdomain name for your installation, like the name for the cluster you give during the Oracle Grid Infrastructure software installation process.

The GNS nameserver entry must resolve to a valid IP address in the public network of the cluster. Request this to be added by your network administrator to the DNS configuration.

```
$ORIGIN cluster1.my-company.com.
@                IN NS gns.cluster1.my-company.com.
gns.cluster1.my-company.com IN A 10.165.111.103
```

Cluster Name	A valid name for your cluster, e.g. cluster1
SCAN Name	Fully qualified name for the SCAN e.g. cluster1-scan.cluster1.my-company.com
SCAN Port	1521
GNS Sub Domain	Subdomain name for GNS e.g. cluster1.my-company.com
GNS VIP Address	IP Address for GNS, e.g. 10.165.111.103
Cluster Node Information	Add all cluster nodes to the configuration. If you use GNS, the virtual hostname and the associated VIP are automatically generated. Just add all the cluster nodes to the configuration by pressing the “Add” button for every node you want to add.

Select “Install and Configure Grid Infrastructure for a Cluster”. Do this even if there is an existing Oracle CRS 10.2 installation. Upgrading an existing installation will require many additional changes which are not documented in this white paper. Only the installation option “Install and Configure Grid Infrastructure for a Cluster” is supported by SAP.

Select the Product Language. We recommend to select the English language.

The Oracle Grid Infrastructure software requires a local ORACLE_HOME directory.

Network Option 2: Use existing DNS without GNS

Note that using this configuration option the SCAN name is resolved through your DNS server. There is no additional subdomain given in the name.

The DNS entry must be present before you start with the installation. It must resolve to three IP addresses. All addresses must belong to the public network of your cluster.

Example DNS entry:

<code>cluster1-scan.my-company.com</code>	<code>IN A 10.165.110.166</code>
	<code>IN A 10.165.110.167</code>
	<code>IN A 10.165.110.168</code>

In this example, the domain qualifier is my-company.com. cluster1-scan is similar to a simple host entry, with the difference that it resolves to three different IP addresses. The Oracle 11.2 client software uses these three addresses in a round robin algorithm to connect to a SCAN listener.

If the SCAN name cannot be resolved, you will see an error.

Cluster Name	A valid name for your cluster, e.g. cluster1
SCAN Name	Fully qualified name for the SCAN e.g. cluster1-scan.my-company.com
SCAN Port	1521
Cluster Node Information	Add all cluster nodes to the configuration. The virtual hostnames can be resolved either through DNS or by host resolution in the /etc/hosts file. Default is to extend the public hostname with the extension -vip, e.g. hostname-vip

	Add all the cluster nodes to the configuration by pressing the “Add” button for every node you want to add.
--	---

If you have an existing Oracle CRS 10.2 installation, you may want to use the existing names for the virtual IP name of the nodes.

All Installations: Specify Network Interfaces

You must identify the network adapters for public and private networks. You need at least a public and a private network. For the public network, which is used for the VIP, the subnet mask must match the subnet mask from the network configuration of the operating system. Otherwise, the virtual IP cannot be bound to the network adapter.

If you have more than two network cards, use only two: one for the private and one for the public network. Specify "Do not use" for any additional network adapters.

Note: If your operating system allows virtual network adapters, make sure that the interface configuration for the public network can hold the VIP on every node.

Note: The names of all network interfaces for the private interconnect as well as the public network adapters must be identical on all nodes. Make sure the system setup on all cluster nodes use the same interface name for public and private network.

Storage Option 1: Place OCR and Voting Files on Oracle ASM

Select Oracle ASM only if you plan to place OCR and voting files on Oracle ASM. This is recommended if you perform a first-time RAC enabling of your single instance SAP database system. It is optional if you upgrade from an existing Oracle RAC 10g enabled SAP system.

Storage Option Information	Automatic Storage Management (Oracle ASM)
Disk Group Name	OCR
Redundancy	High or Normal
Oracle ASM Password	Required for Oracle ASM operation
IPMI	Depending on your platform support
Oracle ASM Database Administrator Group	asmdba

Oracle ASM Instance Administration Operator Group	asmoper
Oracle ASM Instance Administrator Group	asmadmin

Create an Oracle ASM diskgroup with name OCR. This Oracle ASM diskgroup consists of 3 separate disks in normal redundancy mirroring. With high redundancy, you will need 5 disks to build the diskgroup. The OCR diskgroup is only used for storing voting files and OCR repository data. No other data should go to this diskgroup.

If you do not see any disk at all or not all disks you have expected, press the “Change Discovery Path” button and specify the pattern to filter for available disks.

If your OCR and voting information is located on Oracle ASM, you must specify a password for the Oracle ASM administrator. This password should be different from your database password for user account SYS and SYSDBA.

If you place OCR and voting files on Oracle ASM, we recommend to use the fine grained permission model as suggested and used per default by Oracle.

Storage Option 2: Shared Filesystem for OCR and Voting Files

If you upgrade an existing Oracle RAC 10.2 database you should continue to use a shared filesystem location to store OCR and voting files. The use of Oracle ASM is not recommended in this case.

Note: We recommend that you create new files for your Oracle 11.2 Grid installation and do not reuse existing files from previous installations.

Should anything fail during installation, you can easily switch back to the previous configuration without compromising your system integrity.

Storage Option Information	Shared File System
OCR Storage Option	Normal Redundancy

	Specify storage location for 2 OCR files on different filesystems.
Voting Disk Storage Option	Normal Redundancy Specify storage location for 3 voting files on different filesystems

Do not reuse the OCR and voting files from an existing Oracle 10.2 CRS installation.

Choose new files instead. If anything fails during installation and you need to deconfigure the failed installation, you can always switch back to the previous configuration.

The voting disks are a three-way-mirror in normal redundancy configurations. The three voting disks should therefore reside in three separate partitions/filesystems of the shared filesystem.

All installations: ORACLE_BASE for GRID Infrastructure

Oracle Base	/oracle/BASE
Software Location	/oracle/GRID/11202
Voting Disk Storage Option	Normal Redundancy Specify storage location for 3 voting files on different filesystems

The ORACLE_BASE location for the Oracle Grid Infrastructure 11.2 holds diagnostic and metadata information for the local 11.2 grid installation. As the grid software location is local for each cluster node, the ORACLE_BASE location should be local as well. This deviates from the Oracle database software installation, where a shared location for ORACLE_BASE

and ORACLE_HOME must be used. Select an Oracle Base directory residing on a local filesystem. Select a software location residing on a local filesystem.

Inventory Directory	/oracle/oraInventory
---------------------	----------------------

If this is the first installation of Oracle products, you must specify a location for the Oracle inventory. The inventory location must be private for each node.

The Oracle inventory holds metadata information of all installed Oracle software on a node. In our example the inventory location is in the directory /oracle/oraInventory. This directory is local on every node in the cluster. Do not confuse the inventory location with the software installation directory for the Oracle Grid software.

Make sure to specify a path for the inventory that is not shared among the nodes in the cluster. Check that the Oracle inventory can be fully accessed by the OS user performing the installation. In case of Oracle Grid make sure that the inventory directory and all subdirectories are writable by user oracle. On all nodes, do a "chown -R oracle:oinstall /oracle/oraInventory". This check also applies if you install a patchset or a single patch for the Oracle software..

The Oracle Universal Installer checks that all minimum requirements have been met. If this check is not passed, do not continue with the installation!

In case of errors or warnings, cancel the installation,, resolve the root cause of the error, and then repeat the installation. Look for MOS (My Oracle Support formerly known as Metalink) notes on how to completely delete a failed installation for your platform.

The Oracle Clusterware is started at the end of the installation. You can verify this by executing the crsctl command. The output of command “/oracle/GRI/11202/bin/crsctl status resource -t” should confirm that all the defined resources are created successfully.

Depending on your chosen installation option, you will notice resources for Oracle ASM, GNS and some additional VIPs. Note that the GSD service is OFFLINE by default. All the rest should be ONLINE.

```
$ /oracle/GRID/11202/bin/crsctl status resource -t
```



NAME	TARGET	STATE	SERVER	STATE_DETAILS
Local Resources				
ora.LISTENER.lsnr				
	ONLINE	ONLINE	vmkb1	
	ONLINE	ONLINE	vmkb2	
	ONLINE	ONLINE	vmkb3	
	ONLINE	ONLINE	vmkb4	
ora.OCR.dg				
	ONLINE	ONLINE	vmkb1	
	ONLINE	ONLINE	vmkb2	
	ONLINE	ONLINE	vmkb3	
	ONLINE	ONLINE	vmkb4	
ora.asm				
	ONLINE	ONLINE	vmkb1	
	ONLINE	ONLINE	vmkb2	Started
	ONLINE	ONLINE	vmkb3	Started
	ONLINE	ONLINE	vmkb4	Started
ora.gsd				
	OFFLINE	OFFLINE	vmkb1	
	OFFLINE	OFFLINE	vmkb2	
	OFFLINE	OFFLINE	vmkb3	
	OFFLINE	OFFLINE	vmkb4	
ora.net1.network				
	ONLINE	ONLINE	vmkb1	
	ONLINE	ONLINE	vmkb2	
	ONLINE	ONLINE	vmkb3	
	ONLINE	ONLINE	vmkb4	
ora.ons				
	ONLINE	ONLINE	vmkb1	
	ONLINE	ONLINE	vmkb2	
	ONLINE	ONLINE	vmkb3	
	ONLINE	ONLINE	vmkb4	
ora.registry.acfs				
	ONLINE	ONLINE	vmkb1	
	ONLINE	ONLINE	vmkb2	
	ONLINE	ONLINE	vmkb3	
	ONLINE	ONLINE	vmkb4	

Cluster Resources

ora.LISTENER_SCAN1.lsnr				
1	ONLINE	ONLINE		vmkb2
ora.LISTENER_SCAN2.lsnr				
1	ONLINE	ONLINE		vmkb4
ora.LISTENER_SCAN3.lsnr				
1	ONLINE	ONLINE		vmkb3
ora.cvu				
1	ONLINE	ONLINE		vmkb3
ora.gns				
1	ONLINE	ONLINE		vmkb4
ora.gns.vip				
1	ONLINE	ONLINE		vmkb4
ora.oc4j				
1	ONLINE	ONLINE		vmkb3
ora.scan1.vip				
1	ONLINE	ONLINE		vmkb2
ora.scan2.vip				
1	ONLINE	ONLINE		vmkb4
ora.scan3.vip				
1	ONLINE	ONLINE		vmkb3
ora.vmkb1.vip				
1	ONLINE	ONLINE		vmkb1
ora.vmkb2.vip				
1	ONLINE	ONLINE		vmkb2
ora.vmkb3.vip				
1	ONLINE	ONLINE		vmkb3
ora.vmkb4.vip				
1	ONLINE	ONLINE		vmkb4

Oracle Clusterware Parameter Adjustments for SAP Installations

In some cases the default values of several timing related parameters for error detection and health check are too aggressive for a complex SAP environment, especially if the service times of components like network or disk I/O is not deterministic in a sense that there is a high deviation from the mean or average values.

Important timing parameters for the Oracle Clusterware are *misscount*, *disktimeout*, *reboottime* and *diagwait*.

The parameter *disktimeout* specifies the maximum I/O completion time for read/write access to the OCR repository and the voting disks. Timing values are given in seconds. A node must be able to access at least one of the OCR repository files and at least one of the voting disks; otherwise the node will leave the cluster by issuing a reboot.

The parameter *misscount* gives the maximum time in seconds for outstanding communication probes via the network connections among the nodes. It checks network healthiness. If a node can't reach any of the other nodes within *misscount* seconds timeframe, it will start split-brain resolution and probably evict itself from the cluster by doing a reboot.

Parameter *reboottime* holds the estimated time for a reboot. Default value is 3 seconds. It is the time assumed to bring down a node completely.

The parameter *diagwait* is the amount of time to wait in seconds after a node is considered dead and reconfiguration can safely start. This parameter is important in conjunction with the *oproc* timing values. A dead or blocked node must already be fenced out by *oproc* (via a fast reboot) before the remaining nodes can safely start reconfiguration. If reconfiguration starts too early, data integrity may be compromised.

Always check MOS note 294430.1 "CSS Timeout Computation in Oracle Clusterware" for latest information regarding the parameters and their relationship.

You can query the values by the *crsctl* command. Switch to user root, then issue the command

```
# $GRID_HOME/bin/crsctl get css disktimeout
```

to query the current value of the parameter.

Set the parameter by using command

```
# $GRID_HOME/bin/crsctl set css disktimeout 200
```

as an example.

You must restart Oracle Clusterware on all nodes to have the parameters take effect. Easiest way is to reboot all nodes.

In rare cases, the default timing constraints for the Oracle health monitoring processes can't be met due to long or poor scheduling times in the operating system, which may cause false node evictions (Oracle Clusterware will reboot the node).

If you notice suspicious node evictions by Oracle Clusterware you should change the timeout values of the 'diagwait' parameter (see MOS note 559365.1). The Oracle Clusterware stack must be down on all nodes, modification to diagwait should be made using the crsctl command and the Oracle Clusterware should be re-started. Always make sure that the Clusterware 'misscount' is greater than diagwait.

Recommended value for diagwait is 13.

See MOS note 265769.1 "Troubleshooting CRS Reboots" for further information.

Install Oracle RAC 11.2.0.2 Software on a Cluster

Once the Oracle Grid Infrastructure software has been successfully installed and started, the next step in the configuration process is to install the database software.

A major change compared to earlier releases of Oracle Database Software in an SAP environment is that the installation of Oracle RAC 11.2.0.2. is now performed by the user oracle. This is the same user and owner of the software as used for the installation of the Oracle Grid Infrastructure .

If this user already exists, no additional preparation for the installation is required.

Nevertheless, for SAP management purposes, the user ora<sid> is required. If not already configured, this task should be executed now.

Create user ora<sid> on all nodes in the cluster, if this has not already been done. Use a shared home directory on the shared filesystem for user ora<sid>. This makes administrative tasks much easier, as changes to profiles must be performed only once and are valid or visible on all nodes in the cluster. All configuration examples in this paper assume a shared home directory for user ora<sid>.

If the home directory of user ora<sid> already exists on a local filesystem from a previous SAP installation, relocate this directory to a shared home directory on the shared filesystem.

Similarly to the way the user oracle was set up for the CRS software, secure shell access between all nodes in the cluster must be set up as explained below so that they all can use a shared home directory.

Log on to the first node as user ora<sid>.

Create directory ~/.ssh, if it does not exist already. Go to this directory.

```
$ mkdir .ssh  
$ cd .ssh
```

Create private and public key files for RSA and DSA authorization:

```
$ ssh-keygen -t rsa -f <path_to_homedirectory>/.ssh/id_rsa_<nodename>
```

Do not specify a passphrase, just hit Enter when asked.

```
$ ssh-keygen -t dsa -f <path_to_homedirectory>/.ssh/id_dsa_<nodename>
```

Once again, do not specify a passphrase. Hit Enter.

Repeat this on all nodes. Remember that you must log on to the node, as key generation is dependent on the node itself. After all private and public key files have been generated you must create a config file containing pointers to the identity file for every node.

These identity files contain private keys of the respective nodes. You must give the full pathname to these files. Example for the file `~/.ssh/config`:

```
ForwardX11 no
PasswordAuthentication no
Host oracx1
IdentityFile /saphome1/orarac/.ssh/id_rsa_oracx1
IdentityFile /saphome1/orarac/.ssh/id_dsa_oracx1
Host oracx2
IdentityFile /saphome1/orarac/.ssh/id_rsa_oracx2
IdentityFile /saphome1/orarac/.ssh/id_dsa_oracx2
Host oracw1
IdentityFile /saphome1/orarac/.ssh/id_rsa_oracw1
IdentityFile /saphome1/orarac/.ssh/id_dsa_oracw1
Host oracw2
IdentityFile /saphome1/orarac/.ssh/id_rsa_oracw2
IdentityFile /saphome1/orarac/.ssh/id_dsa_oracw2
```

Concatenate all files containing public keys to the file `authorized_keys`

```
$ cat *.pub > authorized_keys
```

Change the file permissions

```
$chmod 600 config
$chmod 600 authorized_keys
$cd
$chmod 700 .ssh
$chmod 755 ~
```

You should now be able to open a ssh connection to all hosts without being prompted for a password. Try this by executing

```
$ssh <nodename> date
```

Repeat this on all nodes in order to have the fingerprints recorded in the file `known_hosts`.

Check the Oracle documentation for all other minimum requirements before starting the installation. Oracle provides you with a tool for verifying that the conditions for a successful installation have been met. This tool is called `cluvfy` and is shipped with the Oracle software.

Note: It is recommended that you use the tool `cluvfy` to verify that all minimum requirements have been met.

Note that you must install the database software in a separate `ORACLE_HOME` location. Do not choose the same location as for the Oracle Clusterware.

Prepare the storage location for storing the shared `ORACLE_HOME` directory in the cluster. The Oracle RDBMS software should be installed into an empty directory, accessible from all nodes in the cluster. We assume that the name for this directory is `/oracle/<SID>/11202`. This directory naming scheme follows the SAP defaults for single instance database installations.

Note: If you already have a single instance Oracle 11.2 installation, rename the existing directory and use an empty directory.

Prepare a directory for `ORACLE_BASE` of the RDBMS installation on the shared filesystem. This directory will be used for holding diagnostic data and must be shared among all cluster nodes for SAP installations.

Note: The ORACLE_BASE for Oracle Database software and Oracle Grid Infrastructure software are different directories. Do not use the same for both installation types with SAP.

We recommend that you use directory /oracle/SID as the location for ORACLE_BASE on a shared filesystem.

This differs from the recommendations for a single instance database installation, where ORACLE_BASE is /oracle. It is a tribute to the fact that the Oracle database software is installed on a shared filesystem and the information stored under ORACLE_BASE should be accessible from all nodes in the cluster.

Log on as user oracle and start the installation with the Oracle Universal Installer. Select “Install database software only” in the step Installation Option.

Install Options	Install database software only
Installation Type	Real Application Clusters database installation
Product Language	English
Database Edition	Enterprise Edition
Oracle Base	/oracle/<SID>
Software Location	/oracle/<SID>/11202
Database Administrator Group	dba
Database Operator Group	oper

Please note: Although you can use the same Oracle Base directory for all installations, we recommend to use a separate Oracle Base location with every Oracle database software installation. For use with SAP, the Oracle Base directory has to be on a shared filesystem.

Use exactly the suggested groups. Remember that the software installation owner is user oracle with primary group oinstall. By choosing dba as the OSDBA and oper as OSOPER group, any user belonging to these groups can perform administrative tasks like starting or stopping the database.

SAP user ora<sid> belongs to group dba and is allowed to perform all administrative tasks.

Note: Software updates, patchsets and single patches are configuration tasks and must be accomplished by user oracle.

Check the summary screen. If all information is correct, press Install to start the installation process.

The Oracle Universal Installer checks that all minimum requirements have been met. If this check is not passed, do not continue with the installation!

Once the software has been installed, you must run the root.sh scripts on all nodes as user root.

The last screen shows the result of the installation. Do not continue with configuration changes if there are any warning messages.

Note: The shared filesystem holding the shared ORACLE_HOME software installation directory should support memory mapped files. Memory mapped files are used by Oracle Clusterware to monitor the health of a running Oracle instance.

If the shared filesystem on your platform does not allow memory mapped files, you must create a symbolic link to a local filesystem, which supports memory mapped files.

The name of the memory mapped file is hc_<INSTANCE_NAME>.dat. The file is located in directory \$ORACLE_HOME/dbs.

After the software installation, create a local directory on a local filesystem on every node. Copy the file \$ORACLE_HOME/dbs/hc_<INSTNAME>.dat to the local directory. Perform this on every node in the cluster.

Now delete the file \$ORACLE_HOME/dbs/hc_<INSTNAME>.dat. On each node of the cluster, create a symbolic link in \$ORACLE_HOME/dbs pointing to the copied file in the local filesystem.

```
ln -s /<local>/hc_<INSTNAME>.dat
$ORACLE_HOME/dbs/hc_<INSTNAME>.dat
```

Prepare the Database Upgrade

The following sections apply only if you upgrade an existing Oracle RAC 10.2 system. If you perform a single instance to Oracle RAC migration, you can skip these steps as your database is already upgraded or installed with Oracle 11.2.0.2. Continue with configuration in section "Database Controlfile Parameters".

Before you can start the database upgrade using DBUA, some preparations need to be performed. As already mentioned, the software owner of the RDBMS software will be changed from user ora<sid>.dba to user oracle:oinstall. We recommend that you re-install the Oracle 10.2 RDBMS software as user oracle:oinstall and also re-apply the patchset 10.2.0.4/5, depending on the actual version you are using.

Pin all cluster nodes in Oracle Clusterware

To enable an Oracle RAC 10.2 installation together with Oracle Clusterware 11.2.0.2, you need to pin the nodes in CRS. This must be done as user root:

```
# /oracle/GRID/11202/bin/crsctl pin css -n oracx1
```

Repeat this for all nodes in your cluster.

Re-install Oracle 10.2 Database Software and Patchsets

Before you start with the re-installation, rename the existing 10.2.0.x \$ORACLE_HOME. Check that the new name is visible on all nodes in the cluster.

Check that user oracle has write permissions for directory /oracle/<SID>.

As user oracle:oinstall, re-install the RDBMS software to the ORACLE_HOME location used in your installation, e.g. /oracle/<SID>/102_64. Perform a software installation only. Start the Oracle Universal Installer:

```
$runInstaller -ignoreSysPrereqs
```

If the OUI complains on the CRS version or other checks, you can ignore this.

Perform a software installation only. Select all nodes from your cluster. Do not create a database. A temporary database will be created after the patchsets are applied.

After the initial installation, re-apply the patchset (10.2.0.4 or later) which you have been using before. Note that this step is required as the upgrade assistant can't upgrade directly from 10.2.0.1.

Next preparation for use of DBUA is a valid registration of the database in the CRS repository. You can achieve this by creating a simple temporary cluster database with DBCA, specifying same database name and instance names as in your old 10g RAC installation.

First, before starting DBCA, use NETCA to configure a listener for the temporary database. From your freshly re-installed 10g ORACLE_HOME start netca with user account oracle:

```
$ export ORACLE_HOME=/oracle/NW4/102_64  
$ /oracle/NW4/102_64/bin/netca
```

Create only a listener configuration for all your nodes in the cluster. Use same port settings as in your previous setup, so it is probably SAP default 1527.

Use the same listener name as in your previous setup, e.g. LISTENER_<SID>

After netca is finished with the configuration, use command `crsctl status resource -t` to check that the listener resources are created and started on all nodes in the cluster.

Now create a simple temporary cluster database with tool DBCA from the freshly re-installed 10g ORACLE_HOME. Use same database name (<SID>).

Start dbca

```
$ /oracle/NW4/102_64/dbca
```

Create a cluster database by selecting all nodes in your cluster. Use the same name as your old database.

Be careful with the SID Prefix for the instances: If your instance-numbers from the old installation have been 3 letters, e.g. <SID>001 for the first database instance, specify <SID>00 for the SID Prefix as dbca will only add a single letter for instance number on the tail.

Global Database Name	<SID>
SID Prefix	<SID>00

Create a custom database without datafiles (only SYSTEM and SYSAUX), and deselect all database options as this database will be created to just register in CRS.

It will not be used at all and the datafiles can be dropped later.

When asked for listener registration, select “Register the database with all listeners”.

These are the listeners you have created with NETCA.

After dbca is finished, the temporary database should be running on all nodes.

Try to connect to the instances: Set environment variables ORACLE_HOME and ORACLE_SID, use `sqlplus / as sysdba` to connect.

Before continuing, shutdown the temporary cluster database.



```
$/oracle/GRID/11202/bin/crs_stop ora.<SID>.db -f
```

Check that all instances are stopped:

```
$/oracle/GRID/11202/bin/crsctl status resource -t
```

If the database is stopped, exchange the database files, switching to your original database:

Copy the server parameter file (spfile<SID>.ora) for the existing database in the dbs subdirectory from the old, renamed \$ORACLE_HOME to the freshly created server parameter file (spfile<SID>00.ora).

Note: You must keep the name spfile<SID>00.ora, as this is the filename and path registered in CRS.

Check that the owner of the replaced server parameter file is oracle:oinstall.

Location for the server parameter file spfile<SID>00.ora can be found in files /oracle/<SID>/102_64/dbs/init<SID><THREAD>.ora.

Change the ownership of database files

Change ownership of all database files, redo-files, control-files, archived redo-files to user oracle:oinstall. Note that if you are using Oracle ASM for storing OCR and voting information, you must change the database file owner to user oracle:asmadmin.

Check that you can startup the original database with the new installed executables:

First test with sqlplus:

```
$export ORACLE_SID=NW4001
$export ORACLE_HOME=/oracle/NW4/102_64
$/oracle/NW4/102_64/bin/sqlplus / as sysdba
```

```
SQL>startup
```

If this test is successful, shutdown the instance and test with CRS command:

```
$/oracle/GRID/11202/bin/crs_start ora.<SID>.db
```

You should see that the database instances come up on all configured nodes.

Check /etc/oratab

Check that the entry in file /etc/oratab for your database is valid on all nodes in your cluster.

Upgrade the Oracle Database using DBUA

We recommend that you use Oracle tool DBUA for the upgrade of your existing Oracle RAC 10.2 database. Logon to the system as user oracle. Start the upgrade assistant.

Available Databases	<SID>
Pre-11gr2 listeners detected	Yes, migrate to current Oracle Home
Recompile Invalid Objects	Yes, Parallelism >= 3
Backup database	No, must be done before CRS installation
Move Datafiles during upgrade	Do Not Move Database Files
Fast Recovery Area	As in 10g RAC installation
Diagnostic Destination	/oracle/<SID>/diag
Listeners	Select: LISTENER [/oracle/GRID/11202]

Select the recompilation option in the upgrade options screen. Backup of your existing database is a prerequisite which you must have completed before any upgrade step was performed.

Don't move the datafiles, especially Oracle ASM is no valid option.

Note: If you plan to migrate your database to Oracle ASM, this must be done in a separate task.

Check SAP note 1550133 on this. Read the two white papers mentioned in the note:

“Moving your SAP Database to Oracle Automatic Storage Management 11g Release 2: A Best Practices Guide” and

“SAP Databases on Oracle Automatic Storage Management 11g Release 2 Configuration Guidelines for UNIX and Linux Platforms”.

Here you find recommendations on the supported procedures as well as naming conventions for Oracle ASM disks and other relevant information.

Specify your flash recovery area if the existing database already used a flash recovery area. Specify the Diagnostic Destination, which is ORACLE_BASE

Select only the listener from the GRID installation.

Once DBUA is finished, you can start and stop the database with tool srvctl.

```
$ /oracle/GRID/11202/bin/srvctl status database -d <SID>
```

Modify Location for Server Parameter File spfile<SID>.ora

During registration of the temporary database you have created to re-register the Oracle 10.2 database in the Oracle Clusterware, the location of the server parameter file was registered in the repository. This needs to be changed before you can delete the temporary database.

Shutdown the database

```
$ /oracle/GRID/11202/bin/srvctl stop database -d <SID>
```

Copy the file spfile<SID>00.ora from the temporary location to /oracle/<SID>/11202/dbs/spfile.ora.

Edit all init<SID><THREAD>.ora files to point to the new location.:

```
SPFILE='/oracle/<SID>/11202/dbs/spfile<SID>.ora'
```

Update information in the Oracle Clusterware:

```
$ /oracle/GRID/11202/bin/srvctl modify database -d <SID>  
-p /oracle/<SID>/11202/dbs/spfile<SID>.ora
```

Check that the database starts up correctly.

```
$ /oracle/GRID/11202/bin/srvctl start database -d <SID>
```

Manual Upgrade of the Database

We do not recommend a manual upgrade of the database. If you decide to perform a manual upgrade using sqlplus, see MOS note 837570.1, “Complete Checklist for Manual Upgrades to 11gR2”.

Note: If you are upgrading from an Oracle RAC 10.2 installation, remember that all the tasks for the upgrade to Oracle 11.2 must be done as for a single-instance environment. For this, you must set initialization parameter cluster_database=false.

If you are still using a pfile init<SID>.ora for instance startup, edit this file with a text editor and make sure the parameter cluster_database is set to false.

In the case of a single server parameter file spfile.ora which is needed by SAP for RAC, change the setting of the parameter using sqlplus:

```
$ sqlplus / as sysdba
SQL> startup nomount
SQL> alter database set parameter cluster_database='false' scope=spfile;
SQL> shutdown immediate;
```

Change the ownership of all database files as described in section “Change the ownership of database files to oracle:oinstall or oracle:asmadmin”

After the Upgrade: Manual or by DBUA

Change the entries in the spfile to use EZCONNECT for registration with the listener processes:

```
$ sqlplus / as sysdba
SQL> startup nomount
```

```

SQL> alter system set local_listener =
      '//node1-vip.cluster1.my-company.com:1521'
      scope = spfile sid = '<SID>001';
SQL> alter system set local_listener =
      '//node2-vip.cluster1.my-company.com:1521'
      scope = spfile sid = '<SID>002';
SQL> alter system set local_listener =
      '//node3-vip.cluster1.my-company.com:1521'
      scope = spfile sid = '<SID>003';
SQL> alter system set local_listener =
      '//node4-vip.cluster1.my-company.com:1521'
      scope = spfile sid = '<SID>004';

SQL> alter system set remote_listener =
      '//cluster1-scan.cluster1.my-company.com:1521'
      scope = spfile sid = '*';

SQL> shutdown immediate;

```

In this example every instance running on a node connects to the local node listener by using the node VIP. The remote listener entry is the same for all instances and uses the SCAN VIP for the connection.

In the template above, GNS name resolution is assumed. If you don't have configured GNS, you don't need to include subdomain information (.cluster1.)

After this modification is in place, you can delete the listener configuration performed with NETCA as preparation for the upgrade with DBCA / DBUA

Apply all required Patches for SAP

Once the OUI has installed the software, you must apply all patches required for running a SAP system. You can download all recommended and certified patches from the SAP marketplace. Check SAP note 1431800 for further information regarding Oracle Database 11g Release 2.

Log on to the system as user oracle. It is important that the environment of the user oracle now reflects the changes to the new Oracle 11g software release. Before starting the database with the new Oracle executables, these variables must be set or corrected as follows:

Variable	Value
ORACLE_HOME	/oracle/<dbSID>/11202
ORACLE_SID	<dbSID><instance number>
ORACLE_BASE	/oracle/<dbSID>

Additionally, the \$PATH and the \$LD_LIBRARY_PATH variable should include the path to the new Oracle executables.

We recommend that you use MOPatch to apply all required patches.

Database Control File Parameters

This section applies to existing single instance databases only. If you start with a freshly installed SAP system or your system is already RAC enabled, you can skip this part.

For the planned RAC enabling of the existing database, the database parameters MAXINSTANCES, MAXLOGFILES and MAXLOGMEMBERS need special attention. These parameters are part of the database control files and were specified during initial database creation. Generally speaking, these parameters may not have been changed since the first installation of the R/3 system.

Note: This section applies to upgrades from an old single-instance database only. If your existing system is already Oracle RAC 10.2 enabled, you can skip this section.

MAXINSTANCES should be at least the number of nodes in the cluster, probably with some room for additional growth in the future.

MAXLOGFILES determines the highest number of redo log file groups that can ever be created in the database. Keeping in mind the conventions used by SAP software for naming the log file groups and members, this value should be set to a number greater than the highest GROUP value for any redo log file group.

MAXLOGMEMBERS specifies the maximum number of members, or identical copies, for a redo log file group. The default value for an SAP installation is 2.

If the current values are not sufficient for the planned configuration, a new set of control files needs to be created. To do this, dump the current settings to a trace file as follows:

```
SQL> alter database backup controlfile to trace;
```

This generates an SQL statement together with hints for generating a new set of control files in the current database trace file. The location of the directory for this file can be determined by SQL*Plus as follows:

```
SQL> show parameter user_dump_dest
```

The default directory for this file is /oracle/<SID>/saptrace/usertrace, if SAP naming conventions are used.

The most recent database trace file contains all necessary information. Save the relevant part of this trace file to a temporary file and modify the parameters to appropriate values.

This command gives a list:

```
ls -lr
```

The most recent trace file is at the end of this list.

Copy the contents into a temporary file and correct the values for MAXLOGFILES, MAXLOGMEMBERS and MAXINSTANCES if necessary

If the current parameters are already sufficient, then you don't have to create a new set of control files. If this is not the case, then you need adjust these parameters. Once you have made the changes, follow the instructions given in the trace file for creating a new set of control files.

Automatic Undo Management with Undo Tablespaces

If you do an upgrade of an existing Oracle RAC 10.2 database, you can skip this section.

Oracle and SAP recommend using undo tablespaces in a RAC environment. The use of rollback segments is deprecated and should be avoided in a RAC configuration with SAP.

Note: This section applies to upgrades from single instance databases only. Skip this section if your existing system is already Oracle RAC 10.2 enabled with undo tablespaces in use.

Prepare to use automatic undo management for an SAP installation with the following naming conventions used by SAP:

PSAPUNDO, PSAPUNDO_002, ... , PSAPUNDO_n, depending on the number of instances in the cluster. Preferably, the datafiles for these tablespaces should be located on separate disks.

Creation of the tablespaces can also be done using the following sample SQL script:

```
create undo tablespace PSAPUNDO datafile
'/oracle/<dbid>/sapdataX/undo/undo.data1 ' size 1000m reuse;
create undo tablespace PSAPUNDO_002 datafile
'/oracle/<dbid>/sapdataX/undo_002/undo_002.data1 ' size 1000m
reuse;
. . . . .
create undo tablespace PSAPUNDO_00n datafile
'/oracle/<dbid>/sapdata4/undo_00n/undo_00n.data1 ' size 1000m
reuse;
```

Note: Leveraging automatic undo management also requires changes to the initialization file of an instance. The entries for the rollback segments need to be deleted or commented out.

Add the lines:

```
*.undo_management = auto
<dbid>001.undo_tablespace = PSAPUNDO
<dbid>002.undo_tablespace = PSAPUNDO_002
<dbid>00n.undo_tablespace = PSAPUNDO_00n
```

to the instance-specific initialization files.

Use of automatic undo management is mutually exclusive with the use of rollback segments for the instances in the cluster. It is not allowed to use a mix of both methods. **Either none or all instances in the cluster must be configured to use automatic undo tablespaces.**

Note: If you decide to use undo tablespaces, you have to drop all the tablespace(s) holding the old rollback segments. You cannot set these tablespaces in *offline* mode, as the database server may hang.

Redo Log Groups and Members

Note: This section applies to upgrades from single-instance databases only. If your existing system is already Oracle 10.2 RAC enabled, you can skip this section.

In a standard SAP installation, there are four groups of Oracle redo log files. By default, each group contains one original and one mirrored redo log file. If redo log files are mirrored with the help of hardware or operating system, then each group will consist of one original redo log file only. The default layout in a single instance database installation is organized as follows:

- GROUP 101 (redo1)
 /oracle/<dbSID>/origlogA/log_g101m1.dbf
 /oracle/<dbSID>/mirrlogA/log_g101m2.dbf
- GROUP 102 (redo2)
 /oracle/<dbSID>/origlogB/log_g102m1.dbf
 /oracle/<dbSID>/mirrlogB/log_g102m2.dbf
- GROUP 103 (redo3)
 /oracle/<dbSID>/origlogA/log_g103m1.dbf
 /oracle/<dbSID>/mirrlogA/log_g103m2.dbf
- GROUP 104 (redo4)
 /oracle/<dbSID>/origlogB/log_g104m1.dbf
 /oracle/<dbSID>/mirrlogB/log_g104m2.dbf

The log files are periodically written from redo log log_g101m?.dbf to redo log log_g104m?.dbf. The redo log in use and the redo log being archived always belong to a different set: GROUP 101 and GROUP 103 belong to set A, GROUP 102 and GROUP 104 belong to set B.

In an Oracle RAC configuration with multiple instances, every database instance needs its own group of redo log files. The naming conventions are changed slightly for these additional log file groups. The second instance (thread2) should use a redo log set A containing GROUP 21 and GROUP 23, and a set B containing GROUP 22 and GROUP 24. The third instance will then use set A and B and so on.

Creating additional redo log files for the new database instances can be performed with a simple SQL script as shown in the following example for an Oracle RAC solution with four database instances:

```
alter database add logfile thread 1 group 11
  ('/oracle/<dbsid>/origlogA/log_g11m1.dbf',
   '/oracle/<dbsid>/mirrlogA/log_g11m2.dbf') size 200M reuse;
alter database add logfile thread 1 group 12
  ('/oracle/<dbsid>/origlogB/log_g12m1.dbf',
   '/oracle/<dbsid>/mirrlogB/log_g12m2.dbf') size 200M reuse;
alter database add logfile thread 1 group 13
  ('/oracle/<dbsid>/origlogA/log_g13m1.dbf',
   '/oracle/<dbsid>/mirrlogA/log_g13m2.dbf') size 200M reuse;
alter database add logfile thread 1 group 14
  ('/oracle/<dbsid>/origlogB/log_g14m1.dbf',
   '/oracle/<dbsid>/mirrlogB/log_g14m2.dbf') size 200M reuse;

alter database add logfile thread 2 group 21
  ('/oracle/<dbsid>/origlogA/log_g21m1.dbf',
   '/oracle/<dbsid>/mirrlogA/log_g21m2.dbf') size 200M reuse;
alter database add logfile thread 2 group 22
  ('/oracle/<dbsid>/origlogB/log_g22m1.dbf',
   '/oracle/<dbsid>/mirrlogB/log_g22m2.dbf') size 200M reuse;
alter database add logfile thread 2 group 23
  ('/oracle/<dbsid>/origlogA/log_g23m1.dbf',
   '/oracle/<dbsid>/mirrlogA/log_g23m2.dbf') size 200M reuse;
alter database add logfile thread 2 group 24
  ('/oracle/<dbsid>/origlogB/log_g24m1.dbf',
   '/oracle/<dbsid>/mirrlogB/log_g24m2.dbf') size 200M reuse;

alter database add logfile thread 3 group 31
  ('/oracle/<dbsid>/origlogA/log_g31m1.dbf',
   '/oracle/<dbsid>/mirrlogA/log_g31m2.dbf') size 200M reuse;
alter database add logfile thread 3 group 32
  ('/oracle/<dbsid>/origlogB/log_g32m1.dbf',
   '/oracle/<dbsid>/mirrlogB/log_g32m2.dbf') size 200M reuse;
alter database add logfile thread 3 group 33
  ('/oracle/<dbsid>/origlogA/log_g33m1.dbf',
   '/oracle/<dbsid>/mirrlogA/log_g33m2.dbf') size 200M reuse;
alter database add logfile thread 3 group 34
  ('/oracle/<dbsid>/origlogA/log_g34m1.dbf',
```

```
    '/oracle/<dbsid>/mirrlogB/log_g34m2.dbf') size 200M reuse;

alter database add logfile thread 4 group 41
  ('/oracle/<dbsid>/origlogA/log_g41m1.dbf',
   '/oracle/<dbsid>/mirrlogA/log_g41m2.dbf') size 200M reuse;
alter database add logfile thread 4 group 42
  ('/oracle/<dbsid>/origlogB/log_g42m1.dbf',
   '/oracle/<dbsid>/mirrlogB/log_g42m2.dbf') size 200M reuse;
alter database add logfile thread 4 group 43
  ('/oracle/<dbsid>/origlogA/log_g43m1.dbf',
   '/oracle/<dbsid>/mirrlogA/log_g43m2.dbf') size 200M reuse;
alter database add logfile thread 4 group 44
  ('/oracle/<dbsid>/origlogB/log_g44m1.dbf',
   '/oracle/<dbsid>/mirrlogB/log_g44m2.dbf') size 200M reuse;
```

The new threads need to be enabled before they can be used.

```
SQL> alter database enable public thread 1;
SQL> alter database enable public thread 2;
SQL> alter database enable public thread 3;
SQL> alter database enable public thread 4;
```

After the files and groups have been created and enabled, the old set of redo log files can be dropped. Query v\$log to see which log file group is current:

```
SQL> select group#, archived, status from v$log;
GROUP# ARC STATUS
-----
101 YES INACTIVE
102 NO CURRENT
103 YES INACTIVE
104 YES INACTIVE
11 NO UNUSED
12 NO UNUSED
13 NO UNUSED
. . . . .
44 NO UNUSED
```



Do as many *log switches* as needed to ensure that the *current log* is in the newly created group. Query v\$log again if in doubt.

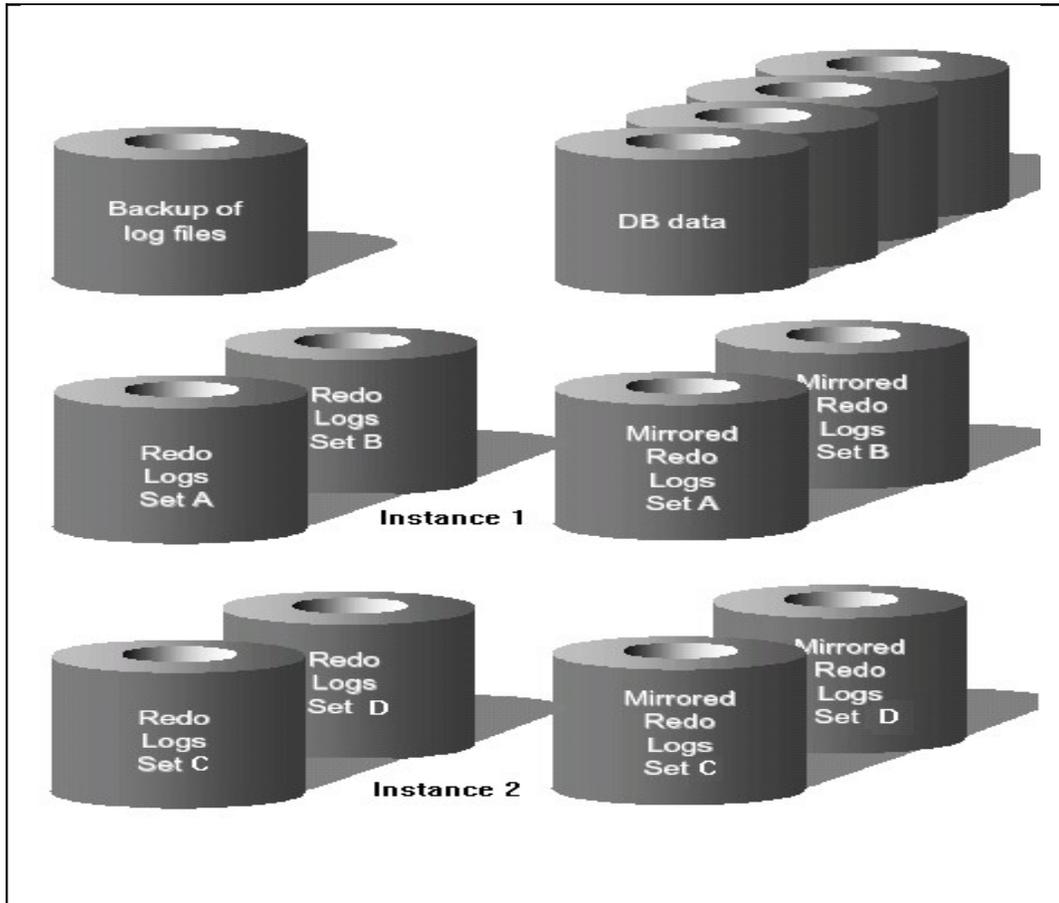
```
SQL> alter system switch logfile;  
SQL> alter system checkpoint;  
SQL> alter system switch logfile;  
SQL> alter system checkpoint;  
SQL> alter system switch logfile;  
SQL> alter system checkpoint;  
SQL> alter system switch logfile;  
SQL> alter system checkpoint;
```

Once you have done this, the old log file groups can be dropped safely:

```
SQL> alter database drop logfile group 101;  
SQL> alter database drop logfile group 102;  
SQL> alter database drop logfile group 103;  
SQL> alter database drop logfile group 104;
```

Don't forget to delete the log files of group 101-104 at the OS-level!

In the given example, it is assumed that the log files are not mirrored by hardware.



SAP recommends that the members of a log file group reside on different disks for performance reasons. The diagram above shows the optimal configuration in which every set of redo logs resides on a separate disk or disk volume.

Modify Oracle Initialization Parameters for RAC

For correct operation of the RAC cluster database environment, the database initialization parameters have to be changed. With a single-instance Oracle database, there is only one parameter file `init<dbSID>.ora` in the directory `$ORACLE_HOME/dbs`. If you are running a RAC cluster database configuration, every database instance needs a separate parameter file with the name `init<dbSID>` or the default `init.ora` file for instance startup. In an SAP environment, SPFILE is now supported, and the modification to use it, is simple to perform.

Note: With a shared `$ORACLE_HOME`, the single server parameter file is stored in a shared location on the shared filesystem.

SAP uses the location `$ORACLE_HOME/dbs` for configuration files as well. The configuration of the `BR*SPACE` utility is stored in this directory. Maintaining these files and the Oracle server parameter file with SAP tools require a shared location. Distributed (local to a node) configuration is not supported.

Oracle and SAP recommend to use an SPFILE in an RAC environment.

Note: The following section applies if you are upgrading an existing SAP installation and you are using a PFILE `init<SID>.ora`.

The original initialization file `init<dbSID>.ora` must be copied from the old directory to the new directory `$ORACLE_HOME/dbs`. If the `IFILE` parameter is used in the old file, then any referenced file within should also be copied from the old location to the new location. Note that if this *include* mechanism was used for database initialization, then it should be avoided in the future. This recommendation is also from SAP and is valid for normal upgrades to Oracle 11g as well. This will ensure an easier transition to binary SPFILES in future releases.

In the planned RAC environment, every Oracle database instance needs a private initialization file `init<dbSID><thread>.ora`. A good practice if you want to stay with instance-specific files is to use this file for the instance-specific settings only and to include the common parameters for all instances in a file named `icommon.ora`.

Note: If you are already using an SPFILE, you may create a human-readable pfile `init.ora` to adjust the configuration using a text editor.

If you modify an existing spfile, use sqlplus commands “alter system set parameter=...”

Example: Set instance number 1 for instance NW4001

```
SQL> alter system set instance_number = 1
      scope = spfile sid = 'NW4001';
```

The instance-specific part of the initialization file should look like this:

```
instance_number = <thread>
thread = <thread>
instance_name = <dbsid><thread>
service_names = <dbsid>, <dbsid><thread>
undo_management = auto
undo_tablespace=<tablespacename>
```

An undo tablespace has to be created in the database for every instance. See section undo tablespaces for an explanation of automatic undo management.

Oracle and SAP recommend to use undo tablespaces in an RAC environment.

The file icommon.ora contains all other initialization parameters, which are valid for all instances. In an RAC environment, you need to pay particular attention to the parameters shown in the following table after upgrading:

Parameter	Value	Comment
db_domain	NOT SET	If set, registering the database in CRS will require -m <domain> parameter
Compatible	11.2	
cluster_database	True	Set this after the upgrade to 11g
remote_login_passwordfile	Exclusive	
remote_os_authent	True	
local_listener	//<node-vip.domain>:1521	Use

		EZCONNECT
remote_listener	//<SCAN-vip.domain>:1521	Use EZCONNECT

Note: Adjust the parameters shown in the previous table after the database upgrade. For the upgrade procedure itself, the old settings must be used.

Once the migration to a RAC-enabled database has finished, you should create a single init.ora initialization file instead of multiple init<dbsid>.ora files. Instance-specific values are tagged by the instance name in front of the parameter. The instance name is resolved from the ORACLE_SID environment variable.

Example from an extract of a single init.ora file:

```
<dbsid>001.instance_number = 001
<dbsid>002.instance_number = 002
<dbsid>001.thread = 001
<dbsid>002.thread = 002
<dbsid>001.instance_name = <dbsid>001
<dbsid>002.instance_name = <dbsid>002
<dbsid>001.service_names = (<dbsid>, <dbsid>001)

*.undo_management = auto

<dbsid><thread>.undo_tablespace=<tablespacename>
```

Any parameter that is valid for all instances is preceded by a “*”. You can omit the “*” for common parameters.

Delete any init<dbsid>.ora or init<dbsid><thread>.ora initialization files from the \$ORACLE_HOME/dbs directory.

Create a binary SPFILE (with filename spfile.ora) by using sqlplus

```
connect "/" as sysdba"

SQL> create spfile='spfile.ora' from pfile='init<dbsid>.ora';
```

Note: Do this after successful migration to RAC.

A binary SPFILE is a prerequisite for using Oracle Enterprise Manager in an RAC environment.

Delete any init<dbid>.ora or init<dbid><thread>.ora initialization files from the \$ORACLE_HOME/dbs directory.

Oracle Password File

A password file is not used in the single-instance SAP R/3 configuration by default, but it is necessary for an RAC environment. The database instance will use this file to identify users with DBA and OPER privileges. To create a password file, log on as user ora<dbid>. If you use any other user account, the password file will be unusable. The file itself resides in the same directory as the database initialization files, which is \$ORACLE_HOME/dbs.

Use this command to create the file:

```
orapwd file=<fname> password=<password> entries=<users>
```

The file parameter specifies the filename and is mandatory. Enter the password of the database user SYS for the password parameter. The entries parameter is optional and denotes the number of distinct users with DBA and OPER privileges.

Enable the Cluster Database and Start all Instances

Note: This section applies to upgrades from single-instance databases only. If your existing system is already Oracle RAC 10.2 enabled, you can skip this section.

Once you have finished upgrading to Oracle RAC 11.2 and completed all modifications to the database as explained in this part, (re)enable the cluster database. In the case of a single-server parameter file spfile.ora which is required by SAP for RAC, change the setting of the cluster database parameter using sqlplus:

```
$ sqlplus / as sysdba

SQL> startup nomount
SQL> alter system set parameter cluster_database='true' scope=spfile;
SQL> shutdown immediate;
```

Log on to every node in the cluster as user ora<sid>. Set the environment variable ORACLE_SID to the unique value of the instance intended to run on the cluster node.

Start the Oracle database instance on each node using sqlplus.

```
$ export ORACLE_SID=<DBSID><THREAD>

$sqlplus / as sysdba
connected to an idle instance
SQL> startup
```

The database and the instances are not yet known to the Oracle Clusterware. You must use sqlplus to start the database instances. Once you have completed the configuration use the tool srvctl for controlling the database instances.

Oracle Network Configuration

Correct setup of the network configuration inside and outside the cluster nodes is vital for the smooth and error-free operation of the SAP R/3 system and various administration tools. Depending on the specific needs of the actual operating environment, the underlying operating system and the preference of other Oracle installations at a certain site, the Oracle network configuration files have several possible locations.

This document assumes that EZCONNECT naming is used for the Oracle RAC 11.2 database. Using this naming method no maintenance of network configuration files listener.ora and tnsnames.ora in directory \$ORACLE_HOME/network/admin is required.

Listener Configuration File for Node and SCAN Listeners

Every cluster node that runs an Oracle database instance must have a local listener process to satisfy the connection requests from clients. This listener process is known as the local node listener.

Note: Starting with Oracle Grid 11.2, only one node-listener should be used. This listener is generated with the installation of the Oracle Grid Infrastructure software. The default port used is 1521. In addition, Oracle Grid Infrastructure requires a SCAN listener, which is running on a subset of the cluster nodes.

All Oracle database instances running on a node automatically connect to the local node listener as well as to the SCAN listener(s) and publish available services. Furthermore, all additional software components like Oracle ASM will use these listeners to publish connection information.

The configuration files for the local node listener and the SCAN listeners are in the home directory of the Oracle Grid software. These listeners have been created and configured during Oracle Grid Infrastructure installation. No additional configuration action regarding the ports used or the configured nodes is required if these listeners are used.

Regardless whether you have upgraded an existing Oracle RAC 10.2 database or you perform the RAC enabling for your system the first time, there is no need to copy the listener.ora files from the old \$ORACLE_HOME/network/admin directory. With the Oracle Grid Infrastructure 11.2.0.2 installation all local listeners as well as the SCAN listener

configuration are completely configured and ready to use. The local node listener as well as the SCAN listeners are controlled by the Oracle Clusterware. Copying existing network configuration files is only required for single instance upgrades without any Oracle Grid component running on the node.

Note: Starting and stopping of listeners should only be done using tool srvctl.

The tool lsnrctl should no longer be used in an Oracle RAC 11.2.0.2 environment to start or stop the local listener

Only for upgraded RAC systems: You cannot reuse your existing configuration for listeners. Configuration files are located in the home directory of the Grid Infrastructure installation, and no longer in \$ORACLE_HOME/network/admin. DO NOT reuse old configuration files.

Check availability of the local node listeners in your cluster:

```
$ srvctl status listener
Listener LISTENER is enabled
Listener LISTENER is running on node(s): oracx1,oracx2,oracw1,oracw2
```

Check availability of SCAN listeners:

```
$ srvctl status scan_listener
SCAN listener LISTENER_SCAN1 is enabled
SCAN listener LISTENER_SCAN1 is running on node oracx2
SCAN listener LISTENER_SCAN2 is enabled
SCAN listener LISTENER_SCAN1 is running on node oracw1
SCAN listener LISTENER_SCAN3 is enabled
SCAN listener LISTENER_SCAN1 is running on node oracw2
```

Listeners should be started and stopped using srvctl command. Do not use the lsnrctl start/stop command to control the listener.

Starting a listener manually:

```
$ srvctl start listener -n <node>
```

Stopping a listener manually:

```
$ srvctl start listener -n <node>
```

Furthermore, the status of the listener can be obtained with the command

```
$ lsnrctl status
```

Beside the automatic registration of the Oracle instances to the listener processes, there is still a SID_LIST for every listener to maintain. The SID_LIST entries in the file listener.ora inform the local node listener about the existence of the instance, even if the instance is not running. This is important for the SAP BR*Tools, to differentiate between a stopped and a non-existing instance.

Log on to the system as user oracle. Add the SID_LIST entry at the end of the file listener.ora in the /oracle/GRID/11202/network/admin directory.

```
SID_LIST_LISTENER =  
(SID_LIST=(SID_DESC=(SID_NAME=<SID>001)  
(ORACLE_HOME=/oracle/<SID>/11202)))
```

Remember that you must do this on every node in the cluster as the Grid installation is local on every node. Put everything in one line, do not use CR/LF or whitespaces.

Network Configuration File tnsnames.ora

Whereas the listener configuration file benefits from automatic service registration of the database instances and can be kept small and handy for all listeners, the configuration file for the clients will have more entries if you decide to reuse an existing setup or want to stay with Oracle network configuration in file tnsnames.ora .

If you want to use simplified EZCONNECT client connections you can skip the configuration of file tnsnames.ora for the database instances.

Using EZCONNECT for SAP application server client connections makes maintenance of local client side configuration files unnecessary. All connect information is maintained in SAP instance profiles.

Option 1: Use EZCONNECT for Clients and Database Instances

If all the clients and the database instances use EZCONNECT connection identifiers you do not need a `tnsnames.ora` configuration files at all.

This will eliminate the administration effort for maintaining `tnsnames.ora` configuration files in and outside of the cluster.

Option 2: Use `tnsnames.ora` for Client Connections

Even if you want to stay with an existing `tnsnames.ora` configuration, you must use EZCONNECT naming in your `(s)profile` and therefore you don't need entries for `local_listener` and `remote_listener` in the `tnsnames.ora` configuration file. Delete these entries if you reuse an existing `tnsnames.ora` file.

The first part contains one generic entry for the database as a whole, and as many net service entries as there are database instances for this database. The general net service name is `<DBSID>`, as used in the various profiles of the SAP R/3 application. The main purpose here is to serve as a last resort or default entry for R/3 work processes or other tools that must connect sporadically to the database without the need for a specific instance or service.

The remaining net service entries in this section contain connect descriptors for the database instances running on the cluster nodes. These entries are mostly required for maintaining the database instance. BR*Tools, for example, can use these entries. Another use for these entries is SAP clients using older OCI clients that are not able to log on to the database using database services.

The second section contains all entries that connect to database services in the database. The clients from the SAP R/3 application server instances should use these net service entries to connect not to a single database or a database instance running on a specific node but to a database service which can be relocated to any instance and node by simple CRS commands.

Note: You must add the domain to any service definition (`SERVICE_NAME=...`), if your database uses `db_domain` parameter. Check with command `"/oracle/GRID/11202/bin/lsnrctl status LISTENER"` if database instances register with domain information set.

The Oracle tool srvctl is used to define services within the database instances. For a description and examples of how to define service entries to the database see the section "Define database services for SAP R/3 application instances".

Example: Database NW4 with 4 instances and 4 service definitions for SAP dialog instances D01, D02, D03 and D04:

```
#####
# Filename.....: tnsnames.ora
# Created.....: created by SAP AG, R/3 Rel. >= 6.10
# Name.....:
# Date.....:
#####

### Section 1: general database service and instances

NW4=
  (DESCRIPTION =
    (ADDRESS=(PROTOCOL=TCP)
      (HOST=cluster1-scan.cluster1.my-company.com)
      (PORT=1521))
    )
  (CONNECT_DATA =
    (SERVICE_NAME = NW4)
    (GLOBAL_NAME = NW4)
    (FAILOVER_MODE =
      (TYPE = SELECT)
      (METHOD = BASIC)
    )
  )
)

NW4001=
  (DESCRIPTION =
    (ADDRESS=(PROTOCOL=TCP)
      (HOST=node1-vip.cluster1.my-company.com)
      (PORT=1521))
    )
  (CONNECT_DATA =
    (SID = NW4001)
    (GLOBAL_NAME = NW4)
  )
)
```

```

)
NW4002=
  (DESCRIPTION =
    (ADDRESS=(PROTOCOL=TCP)
      (HOST=node2-vip.cluster1.my-company.com)
      (PORT=1521))
    (CONNECT_DATA =
      (SID = NW4002)
      (GLOBAL_NAME = NW4)
    )
  )
)

NW4003=
  (DESCRIPTION =
    (ADDRESS=(PROTOCOL=TCP)
      (HOST=node3-vip.cluster1.my-company.com)
      (PORT=1521))
    (CONNECT_DATA =
      (SID = NW4003)
      (GLOBAL_NAME = NW4)
    )
  )
)

NW4004=
  (DESCRIPTION =
    (ADDRESS=(PROTOCOL=TCP)
      (HOST=node4-vip.cluster1.my-company.com)
      (PORT=1521))
    (CONNECT_DATA =
      (SID = NW4004)
      (GLOBAL_NAME = NW4.WORLD)
    )
  )
)

### Section 2: entries for SAP application instances

NW4_D01=
  (DESCRIPTION =
    (ADDRESS=(PROTOCOL=TCP)
      (HOST=cluster1-scan.cluster1.my-company.com)
      (PORT=1521)

```

```

)
(CONNECT_DATA =
  (SERVICE_NAME = NW4_D01)
  (GLOBAL_NAME = NW4)
  (FAILOVER_MODE =
    (TYPE = SELECT)
    (METHOD = BASIC)
  )
)
)
)
NW4_D02=
  (DESCRIPTION =
    (ADDRESS=(PROTOCOL=TCP)
      (HOST=cluster1-scan.cluster1.my-company.com)
      (PORT=1521)
    )
    (CONNECT_DATA =
      (SERVICE_NAME = NW4_D02)
      (GLOBAL_NAME = NW4)
      (FAILOVER_MODE =
        (TYPE = SELECT)
        (METHOD = BASIC)
      )
    )
  )
)
)
)
NW4_D03=
  (DESCRIPTION =
    (ADDRESS=(PROTOCOL=TCP)
      (HOST=cluster1-scan.cluster1.my-company.com)
      (PORT=1521)
    )
    (CONNECT_DATA =
      (SERVICE_NAME = NW4_D03)
      (GLOBAL_NAME = NW4)
      (FAILOVER_MODE =
        (TYPE = SELECT)
        (METHOD = BASIC)
      )
    )
  )
)
)

```

```

)

NW4_D04=
(DESCRIPTION =
  (ADDRESS=(PROTOCOL=TCP)
    (HOST=cluster1-scan.cluster1.my-company.com)
    (PORT=1521)
  )
  (CONNECT_DATA =
    (SERVICE_NAME = NW4_D04)
    (GLOBAL_NAME = NW4)
    (FAILOVER_MODE =
      (TYPE = SELECT)
      (METHOD = BASIC)
    )
  )
)
)

```

Copy the tnsnames.ora File to the Location given in TNS_ADMIN

From Oracle Database 10g Release 2 on, SAP software for the ABAP stack uses instant client libraries for database connections. As instant client configurations do not require a \$ORACLE_HOME directory, the location of the network configuration file tnsnames.ora to be used by the client is obtained from the environment variable TNS_ADMIN.

Note: SAP uses environment variable TNS_ADMIN to specify the location of tnsnames.ora. The value of TNS_ADMIN may differ on SAP application server nodes outside of the cluster.

Copy the file tnsnames.ora to all SAP application servers to the location given in the environment variable TNS_ADMIN for user sap<sid> on those servers.

For SAP application instances running within the cluster, you can modify the value of TNS_ADMIN to point to the shared Oracle Home directory holding the network configuration.

Service Registration in CRS for the RAC enabled SAP Database

The idea of database services for SAP application servers is new since Oracle RAC 10g Release 2 and CRS. These database services are defined and controlled in the HA framework of CRS by resources in the same way as databases, instances, listeners etc., for example. To make use of this concept in an SAP R/3 system, the various software components and programs must be known within Oracle CRS, the central framework for starting, stopping and relocating all of these services within the cluster. In terms of CRS, a database service is a resource with specific attributes.

Configure the SAP Database and the RAC Instances in OCR

Before any service entries to the database can be added, the database itself and the instances belonging to the database need to be published to OCR. All configuration tasks are performed using the Oracle Server Control Utility SRVCTL.

Add the SAP database to the OCR configuration. Example for <SID>=NW4

```
$ srvctl add database -d NW4 -o /oracle/NW4/11202 -r PRIMARY
-y AUTOMATIC -s OPEN -t NORMAL
-p /oracle/NW4/11202/dbs/spfileNW4.ora
```

The next step is the addition of all database instances to the configuration. In the example below, 4 instances are added:

```
$ srvctl add instance -d NW4 -i NW4001 -n node1
$ srvctl add instance -d NW4 -i NW4002 -n node2
$ srvctl add instance -d NW4 -i NW4003 -n node3
$ srvctl add instance -d NW4 -i NW4004 -n node4
```

Once the database and the instances are known to the Oracle clusterware, starting and stopping the database should only be performed by using the srvctl command. Use of sqlplus is not recommended.

Example: Starting the database

```
$ srvctl start database -d NW4
```

This starts all instances as well.

Example: Stopping a single instance:

```
$ srvctl stop instance -d NW4 -i NW4001
```

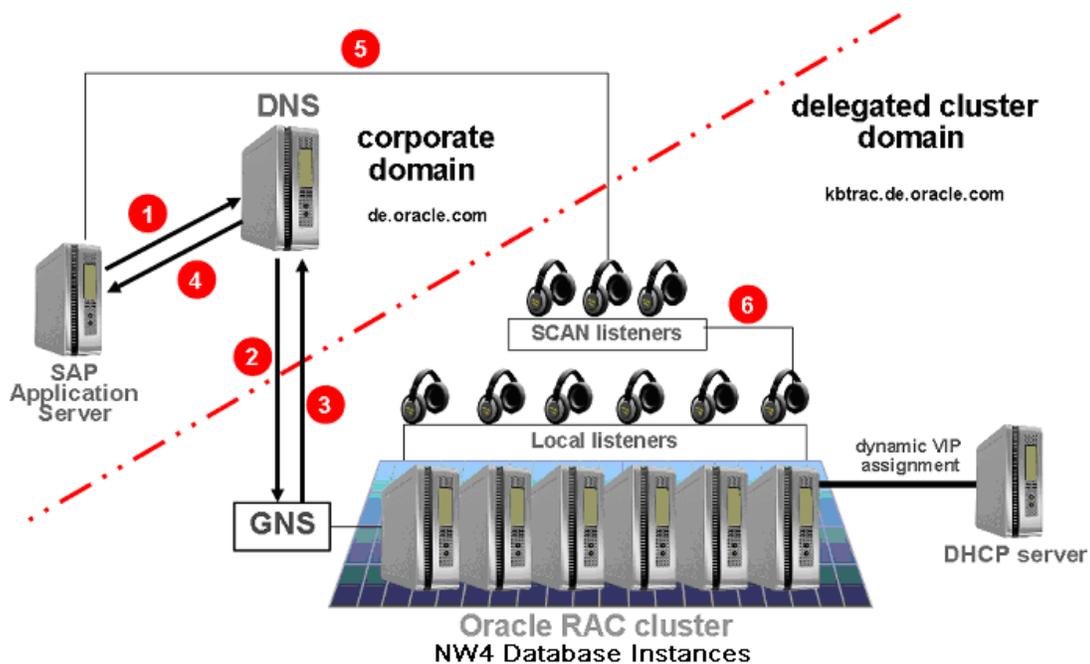
The status of the database and instances can be obtained by

```
$ srvctl status database -d NW4 -v
```

If there is a problem with the database like media corruption, recovery problems etc., srvctl command will not report these kind of problems to the user. So if you notice that the database or some instances does not start correctly, check the alert log of the instances. If the database or some database files need to be repaired by a media recovery, you must perform all administrative task with sqlplus.

Define Database Services for SAP R/3 Application Instances

The idea behind database services definition is to have various SAP application instances connected to arbitrary Oracle database services instead of being connected to a specific instance. The database instances in turn make the configured services available to the application instances. The rules for the placement of the database services are defined within CRS. In a basic setup, every SAP application instance uses a unique database service for connection to the database.



In the picture above, SAP instance running on the SAP application server finds the connection information in the value of parameter `db/ora/tnsname` in instance profile NW4.

This is either an EZCONNECT string or an entry within file `tnsnames.ora`. With this information the SAP instance looks up the connection IP information for the cluster via DNS and GNS hostname resolution. The SAP Instance then connects to the SCAN-Listener using a unique service name in the connection descriptor. The SCAN-Listener forwards the connection request to a local node Listener, to the node where the Oracle database instance offers the requested database service.

Database services for SAP instances are defined within CRS through the `srvctl` command.

Example: Definition of a database service (`-s`) NW4D01 for database (`-d`) NW4 for SAP application instance D01. This database service is configured to be available on database instance (`-r`) NW4001, with the ability to fail over to database instances (`-a`) NW4002, NW4003 or NW4004:

```
$ srvctl add service -d NW4 -s NW4_D01 -r NW4002 -a
NW4001,NW4002,NW4003 -P BASIC -y AUTOMATIC -q true -j long -e
SELECT -m BASIC -z 3 -w 5
```

Note: The name of the service must match the service name given in the connect descriptor in tnsnames.ora if you do not use EZCONNECT. If you use EZCONNECT, the servicename is given directly in the connect string

See the section "Network configuration file tnsnames.ora" for configuring network services in tnsnames.ora if not using EZCONNECT.

Note: The service description should contain the name of the database (NW4) and the name of the SAP instance (D01)

If all these naming conventions are followed, more than one SAP system and Oracle database can run on the cluster without conflicting with each other.

The command given above creates a new resource in CRS with resource name ora.nw4.nw4_d01.svc. NW4001 is the name of the instance running on node oracx1.

This was defined with the "srvctl add instance ..." command above. It is the instance running the service NW4_D01 as preferred instance. The status and attributes of this CRS resource for the service can be obtained by the command *crsctl status resource ora.nw4.nw4_d01.svc -f*

Use SRVCTL to define as many services to the database as there are SAP R/3 application instances configured for the SAP system. The example below shows additional services for the SAP instances D02, D03 and D04:

```
$ srvctl add service -d NW4 -s NW4_D02 -r NW4002 -a
NW4003,NW4004,NW4001 -P BASIC -y AUTOMATIC -q true -j long -e
SELECT -m BASIC -z 3 -w 5
$ srvctl add service -d NW4 -s NW4_D03 -r NW4003 -a
NW4004,NW4001,NW4002 -P BASIC -y AUTOMATIC -q true -j long -e
SELECT -m BASIC -z 3 -w 5
$ srvctl add service -d NW4 -s NW4_D04 -r NW4004 -a
NW4001,NW4002,NW4003 -P BASIC -y AUTOMATIC -q true -j long -e
SELECT -m BASIC -z 3 -w 5
```

SAP User Environment

To test the changes to the SAP Instance and START profiles, modify the files `.dbenv_<hostname>.sh` and `.dbenv_<hostname>.csh` by putting the entry for `db_s_ora_tnsname.in` in comments. Please make copies of the files `.dbenv_<hostname>.sh` and `.dbenv_<hostname>.csh` before making changes.

Note: If you need to run R3trans or tp directly from the command line, you must set `db_s_ora_tnsname`. These tools require the environment variable. Nevertheless, you should test your configuration as shown below. This assures that the changes to the script `startsap` are working correctly.

Example: `.dbenv_<hostname>.csh`

```
...
setenv dbms_type      ORA
# setenv db_s_ora_tnsname $DBSID
setenv ORACLE_PSRV   NW4
setenv ORACLE_SID    $DBSID
setenv DB_SID        NW4
...
```

Example: `.dbenv_<hostname>.sh`

```
...
dbms_type=ORA; export dbms_type
# db_s_ora_tnsname=$DBSID; export db_s_ora_tnsname
ORACLE_PSRV=NW4; export ORACLE_PSRV
ORACLE_SID=$DBSID; export ORACLE_SID
DB_SID=NW4; export DB_SID
...
```

SAP Instance Profile

In the profile directory `/usr/sap/<SID>/SYS/profile`, add an entry in every SAP profile of the configured SAP R/3 instances: Either give the EZCONNECT string specifying the database

service directly or give the service name entry given in the tnsnames.ora network configuration file for this application instance by defining the parameter db/ora/tnsname.

Every SAP instance can be configured to use a specific Oracle net service entry, which in turn resolves to a database service for the application instance. In order to be able to use the service concept and manage the database services for the SAP application instances using CSR commands, the SAP instance must contain the name of the Oracle net service given in the file tnsnames.ora.

Example if using EZCONNECT: Sapsystemname is NW4; Instance name is D01 running on host oracx1. Filename of instance profile is NW4_D01_oracx1

```
...
SAPSYSTEMNAME=NW4
INSTANCE_NAME=D01
db/ora/tnsname=//cluster1-scan.cluster1.my-company.com/NW4_D01
...
```

In the file tnsnames.ora, a net service entry for an SAP application instance is formed as follows: <SID><SERVICENAME>.<DOMAIN>. This value must be given in the profile parameter db/ora/tnsnames. The profiles for the SAP application instances are located in directory /usr/sap/<SID>/SYS/profile.

Example if using tnsnames.ora configuration file: Sapsystemname is NW4; Instance name is D01 running on host oracx1. Filename of instance profile is NW4_D01_oracx1

```
...
SAPSYSTEMNAME=NW4
INSTANCE_NAME=D01
db/ora/tnsname=NW4_D01
...
```

Note: The sapuser environment parameter db_ora_tnsname overrides the parameter setting in the profile for an instance.

The environment variable db_ora_tnsname will be set to the appropriate name for a specific SAP instance in the START profile for that instance.

SAP START Profile

All SAP application instances will be started by the script startsap in a standard installation. In order to be able to use dedicated database services for SAP instances, the START profile of the SAP instances require small changes to interact with the Oracle network setup. The sapstart script sets the environment variable `db_s_ora_tnsname` to the database SID, which is the SAPSID as well, for checks to determine whether the database is already running. Just before the SAP instance is started by the sapstart executable, the environment variable `db_s_ora_tnsname` is set to the value given user environment, or unset if not specified there. If the environment variable `db_s_ora_tnsname` is set, it overrides the setting in the instance profile parameter `db_s/ora/tnsname`. Nevertheless, the parameter `db_s_ora_tnsname` is required for programs and tools like R3trans or tp. So just unsetting this variable in the user environment is not a complete solution.

Note: The user environment parameter `db_s_ora_tnsname` is required to run programs R3trans or tp directly from the command line. Unset this parameter only if you do not run these programs from the command line or for testing purpose to verify that the changes in the START profiles work as expected.

Example if using EZCONNECT: Sapsystemname is NW4, Instance name is D01 running on host oracx1. Filename of instance START profile is START_D01_oracx1

```
. . .  
SETENV_02 = LIBPATH=$(DIR_LIBRARY) :%(LIBPATH)  
EZCONNECT =//cluster1-scan.cluster1.my-company.com/NW4_D01  
SETENV_03 = db_s_ora_tnsname=$(EZCONNECT)  
. . .
```

Example if using tnsnames.ora configuration file: Sapsystemname is NW4, Instance name is D01 running on host oracx1. Filename of instance START profile is START_D01_oracx1

```
. . .  
SETENV_02 = LIBPATH=$(DIR_LIBRARY) :%(LIBPATH)  
SETENV_03 = db_s_ora_tnsname=NW4_D01  
. . .
```

SAP J2EE Server Configuration for RAC Database

This section is valid for Java Add-In installations (JAVA + ABAP stack) as well as for Java Standalone installations of SAP NetWeaver.

The JVM (Java Virtual Machine) by SAP uses the thin JDBC driver from Oracle to connect to the database. Therefore the Oracle network configuration for the Java stack is different from the ABAP stack. With the thin JDBC driver the network configuration files `tnsnames.ora`, `sqlnet.ora` and `listener.ora` are not used. The URL which defines the connect description is stored in an encrypted flat file in the filesystem. The content of this file is also stored in the database. The content of this configuration file is valid for all Java Server Engines in an SAP system. The configuration file is shared via the `sapmnt` access path.

The path to the configuration tool in a mixed ABAP+JAVA installation is `/usr/sap/<SID>/<CI>/j2ee/configtool`. As user `<sidadm>`, go to this directory and start the configuration tool `configtool.sh` from there:

```
$ cd /usr/sap/<SID>/<CI>/j2ee/configtool
$ ./configtool.sh
```

If you see the popup screen asking for the use of default connection settings, answer “yes” to use the default settings. You will then see the starting screen of the “J2EE Engine Config Tool”

In the starting screen, navigate to the “secure store” icon in the navigation tree on the left side.

Look for a key “`jdbc/pool/<SID>/Url`”. This key holds the connect descriptor for the thin JDBC driver. For Oracle RAC use a EZCONNECT connection:

```
jdbc:oracle:thin@//cluster1-scan.cluster1.my-company.com/NW4_J2EE
```

The SAP J2EE engine will use the database service “`NW4_J2EE`” in the example given. The connection request will be serviced by one of the Oracle database instances providing

database service “NW4_J2EE”. If more than one database instance is eligible for the connection, the listener process makes a decision depending on the workload of the database instances.

You must specify the database domain “WORLD” in the connect descriptor if the parameter file of the database (spfile) has the parameter `db_domain` set to the default domain “WORLD”. You can check this by issuing the command “`lsnrctl status LISTENER`” and see if the domain qualifier is part of the service names. The database services for the SAP J2EE instances must be already defined when you want to check with the `lsnrctl` command, see below.

Note: The configuration in the secure store is valid for all SAP J2EE instances. So all SAP J2EE instances will use the same database service. This is different from the ABAP configuration as an ABAP instances uses a private, dedicated database service for the instance.

Define Database Services for J2EE Instances

The database service for SAP J2EE instances is defined with the `srvctl` command in the same way as for the ABAP instances: The rules for the placement of the database services are defined within CRS.

Example:

Definition of a database service (`-s`) NW4_J2EE for an SAP J2EE instance that is configured to be available on database (`-d`) NW4 with required instance (`-r`) NW4001 and NW4002, and with the ability to failover to database instances (`-a`) NW4003 or NW4004:

```
$ srvctl add service -d NW4 -s NW4_J2EE -r NW4001,NW4002
-a NW4003,NW4004 -P BASIC -y AUTOMATIC -q true -j long
-e SELECT -m BASIC -z 3 -w 5
```

Note: The name of the service must match the service name given in the jdbc URL in the secure store configuration. The service name is stated without the domain qualifier “.WORLD”. This qualifier is automatically added by the ORACLE database instances when the database service is published to the listener processes.

The workload distribution for the SAP J2EE engines is managed completely by the database. There is no explicit connection balancing like with the ABAP work processes. Instead, “server side load balancing” is used if the service is defined to run on more than one database instance.

This type of configuration provides the capability to evenly distribute all connections of a single SAP J2EE server engine among all those database instances for which the database service is defined. This connection scenario is different from the ABAP stack as the workload from the J2EE engine is completely random and therefore cannot be controlled as compared with the ABAP stack.

Note: You can configure different services for ABAP and Java application servers, normally running on dedicated nodes in the cluster. By this, you are able to separate the ABAP and Java workload.

SAP BR*Tools

The SAP BR*Tools for Oracle database administration support a RAC configuration. Always download the latest version of the BR*Tools from the SAP service marketplace when using Oracle RAC 11.2.0.2. All required configuration changes for an SAP system with RAC are documented in chapter 6 of the SAP manual "SAP Database Guide: Oracle".

Older versions of BR*Tools store configuration information in directory \$ORACLE_HOME/dbs. The SAP user ora<SID> must be able to write to this directory and perform housekeeping in directory \$ORACLE_HOME/rdbms/audit. This is no longer possible with Oracle default software owner oracle:oinstall.

As a temporary workaround, you can add user ora<SID> to group oinstall.

Still some restrictions on backup and restore apply. Full support is planned for end of Q2CY2011. Check out SAP note 1550133 for updates.

Implementation and Configuration of SAP Enqueue Replication

All setup and configuration steps defined in this document are in line with the implementation of the Oracle Clusterware Agent SAPCTL for SAP Enqueue Replication.

For details on how to configure SAP Enqueue Replication with Oracle Clusterware please consult the Oracle white paper

"Providing High Availability for SAP Resources"

You can obtain a copy of the white paper either from the SAP Developer Network (SDN) website

<http://www.sdn.sap.com/irj/sdn/ora> -> SAP on Oracle Real Application Clusters

or from the Oracle website

<http://www.oracle.com/sap> -> ORACLE RAC FOR SAP -> Best Practices



Configuration of SAP NetWeaver for Oracle
GRID Infrastructure 11.2.0.2 and Oracle Real
Application Clusters 11g Release2
March 2011
Author: Kurt Brög
Contributing Authors:
Jan Klokkers, Markus Breunig

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



Oracle is committed to developing practices and products that help protect the environment

Copyright © 2009, Oracle and/or its affiliates. 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 is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective



