SAP NetWeaver and Oracle Exadata Database Machine

Technical Guide for installation, migration and configuration

Version 1.5

Preface

This document explains all the necessary steps to setup an SAP system based on the SAP NetWeaver technology using the Oracle Exadata Database Machine. All SAP products based from SAP NetWeaver 7.0 on are certified to use the Oracle Exadata Database Machine.

The paper describes the required Oracle software environment settings on the database nodes (Appendix 1 lists a working example which should be followed for an Oracle Exadata deployment for SAP), SAP specific database requirements, information on how to install SAP required database patches to the database nodes, suggestions for the implementation of shared file systems for SAP installations and how to install, configure, manage and control the SAP central services on the database nodes through Oracle Clusterware and its service program SAPCTL.

The Oracle Exadata Database Machine is used for storing the databases of the individual SAP systems. The Oracle Exadata Database Machine cannot be used to run SAP Instances. SAP Instances have to run on separate machines which use the Ethernet or InfiniBand network to exchange data with the database(s) on the Exadata Database Machine. In SAP terminology this is called a three tier architecture. This flexible three tier architecture allows for any combination of hardware and operating systems running the SAP instances to be used with the Oracle Exadata Database Machine. So for instance you can run SAP Application servers on AIX or HP-UX against the Oracle Exadata Database Machine. This flexibility allows an easy introduction of the Oracle Exadata Database Machine in existing SAP environments as it leaves the SAP layer unchanged. On the database nodes of the Oracle Exadata Database Machine you can choose between the Oracle Solaris 11 or Oracle Linux 5 operating system. The only SAP components which are supported to run on the database nodes of the Oracle Exadata Database Machine are the SAP database administration tools (BR*Tools) and the SAP central services (SCS and ASCS). There is support by SAP through SAPINST starting from version 7.0 EP 3 and 7.3 EP 1. Please refer to SAP OSS note 1619343 "SAPinst for Oracle Exadata on Oracle Linux and Solaris X86" to install a new SAP system using the Oracle Exadata Database Machine as the database backend. Databases from already installed SAP systems have to be migrated from existing database servers to the Oracle Exadata Database Machine.

No changes to the standard database schema of the SAP database should be done when being migrated to the Oracle Exadata Database Machine. Changes should not be done to the table and/or index design, the partitioning concept or storage attributes of tables, indexes and partitions. The standard schema of the SAP database is very well designed, tested and proven with thousand's of customers. In addition many SAP administration, monitoring and upgrade tasks depend on the standard database schema layout. Any change to the standard SAP database schema therefore has to be discussed with SAP and an SAP support call should be opened.

SAP OSS note 1590515 "SAP Software and Oracle Exadata" will be updated on a regular base to reflect any changes on using SAP applications with the Oracle Exadata Database Machine.

Overall this documentation complements the existing standard documentation on the Oracle Exadata Database Machine and therefore it is assumed that the reader is familiar with the standard Oracle Exadata documentation.

To understand the requirements and steps outlined in this document it is necessary that the reader is also familiar with the SAP specific support notes and white papers on Oracle RAC ("Configuration of SAP NetWeaver for Oracle Grid Infrastructure 11.2.0.2 and Oracle Real Application Clusters 11g Release 2: A Best Practices Guide"), Oracle ASM ("SAP Databases on Oracle Automatic Storage Management 11g Release 2: Configuration Guidelines for Unix and Linux Platforms"), Oracle Linux and SAPCTL ("Providing High Availability for SAP Resources with Oracle Clusterware 11g Release 2"). All these white papers are stored on the SAP Comunity Network (SCN) under http://scn.sap.com/community/oracle . The SAP Notes are available through the SAP Support Portal for authorized users.

Architecture overview

Short list of key parameters for migration to Exadata

- Database files on ASM only

- no shared file system offered by Exadata (Database File System DBFS is not supported in SAP environment)

- local Oracle Homes for GRID and for RDBMS (exception to standard SAP RAC setup)

- all Oracle software installed on Exadata using Exadata standards

- all Oracle software installed under OS environment 'oracle'

- ASM using standard Exadata setup (different to SAP ASM setups)

- ASM in cluster mode

- SAP instances cannot run on Exadata (Exceptions: ASCS, SCS, ERS)

- SAP central services (ASCS, SCS and ERS) on the database nodes require an SAP Unicode kernel (see chapter "Protection of SAP Central Services" for restrictions and implementation details)

- BR*Tools can be installed on the database nodes

- If non-SAP and SAP databases are operated on the same Exadata, refer to SAP OSS note 1677978 "Administration of mixed SAP/Non-SAP Environments on Exadata"

- Exadata systems are delivered having multiple database- and multiple storage nodes. The database nodes are forming a GRID cluster, which enables RAC or multi instance databases. But RAC is not a must. Also single instance database are supported, which may make profit of the massive parallel IO offered by Exadata.

  Note: The SAP support issue for "Tainted Linux Kernel" disappeared. Exadata Linux kernel is supported by SAP now. SAP OSS note 1634767 "Tainted Linux

Kernel generated by ACFS modules on Exadata" was completely rewritten, now called 1634767 "Support for SAP software in Linux / Oracle ASM/ACFS Cluster".

Prerequisites

Here is a checklist for migration to Exadata. It should carefully checked before starting the migration.

- Read or be aware of additional documentation (SAP OSS note 1590515 "SAP Software and Oracle Exadata", SAP OSS note 1431798 "Oracle 11.2.0: Database Parameter Settings")

- All Oracle software installed on Exadata using Exadata standards, SAP general standards don't apply on this platform

- All Oracle software installed under OS environment 'oracle'

- SAP Kernel CD (minimum 7.00 EP 3 or 7.20 EP 1 and higher)

- BR*Tools software (minimum 7.20 patch 18 and higher)

- Additional disk space for migration and/or network connection between old source system and target Exadata should be available

- SSH-Connection between old source system and target Exadata should be available (user 'oracle')

- Exadata parameter should be set in Oracle init<SID>.ora
  Please check the Best Practice Parameter Reference for actual values!

  ○ The parameter log_buffer should be checked on Exadata for Data Guard configurations. Although the parameter log_buffer is currently not listed in the SAP OSS note 1431798 "Oracle 11.2.0: Database Parameter Settings" for Exadata systems, in a Data Guard configuration, please make sure that the value of log_buffer is not less than 128M and may be set explicitly. This ensures adequate buffer space for new LGWR transport.

  ○ Verify parameter disk_asynch_io having the default value of TRUE, which enables asynchronous IO in ASM.

  ○ Parameter filesystemio_options should be set to SETALL although it does not affect RDBMS IO performance directly because using ASM there is no underlying file system layer. Nevertheless is SETALL recommended because it enables asynchronous IO for RDBMS utilities like RMAN or Data Pump, which may write to file systems extensively. See MOS notes 120697.1 "Init.ora Parameter 'FILESYSTEMIO_OPTIONS' Reference Note" and 751463.1 "ASM Inherently Performs Asynchronous I/O Regardless of filesystemio_options Parameter" for more information.

  ○ Fast recovery area is not generally recommended by SAP mainly not for backups due to no support by BRARCHIVE and BRBACKUP. It can be used for flashback logs of database flashback feature (if activated).

- Ensure that database character set is the same as on old source database

- For the time of a O2O Migration change to global_names=false if SAPSID is the same on old source and new target

ASM Disk Group Recommendations for SAP Databases

On Exadata ASM is already preinstalled in a cluster fashion together with RAC on all database nodes. This particular ASM on Exadata setup has some differences compared to regular Single-Instance or RAC on ASM installations. These differences should be pointed out here shortly before starting with ASM disk setup.

ASM Environment

SAP OSS note 1550133 "Oracle Automatic Storage Management (ASM)" describes the requirements to run ASM in SAP environments. Exadata is included in the OSS note because it only supports ASM for database storage. The main difference between regular ASM setups and Exadata is: ASM on Exadata doesn't support ACFS. For this reason NFS offered outside of Exadata or an external ZFS appliance could be used with Exadata to provide shared file system storage. This shared storage is only used by SAP components, so it doesn't negatively impact Exadata.
Please refer to the above OSS note for version requirements for SAP kernel, BR*Tools and sapinst if used on ASM.

All Exadata database nodes come with an ASM instance already configured, so there is no need for ASM installation.

An Oracle Whitepaper 'Moving your SAP Database to Oracle Automatic Storage Management 11g Release 2, A Best Practices Guide' describes all the necessary steps for ASM disk setup in detail. Unfortunately it deals with single instance setups only but Exadata is operating a Real Application Cluster RAC, which also includes ASM. An ASM plus RAC setup has some minor differences, which must be kept in mind, when reading the whitepaper. An ASM on RAC setup requires each ASM instance to have a unique SID within that cluster or Exadata, so database node 1 is usually operating ASM instance '+ASM1' and so forth. Exadata and other RAC setups are using the OS user 'oracle' for installing Oracle software and this OS user must also been taken for administration. To adopt the user environment to the respective Oracle part, ASM or RDBMS, please refer to SAP OSS note 1554661 "Configuration of environment for 'oracle' user" and 1598594 "BR*Tools configuration for Oracle inst. under 'oracle' user".

ASM redundancy

Although there are no specific requirements for ASM Disk Groups storing the SAP databases on the Oracle Exadata Database Machine, it is a best practice to use a redundancy level of high for a production SAP database to achieve the highest level of protection against any type of storage failure. Other SAP databases used for development, test and QA may use a normal ASM redundancy level. There is no additional need for storage based replication, if normal or high redundancy is used.

Normal redundancy means, the database writes all data twice to two different locations (fail groups of an ASM disk), whereas high redundancy means the database writes all data 3 times to 3 locations (fail groups).

For a good overview of ASM redundancy feature, please have a look into Oracle whitepaper 'SAP with Oracle Real Application Clusters 11g Release 2 and Automatic Storage Management' or Oracle documentation.

Setup ASM Disk Groups

In line with the standard Oracle Exadata setup you should have at least one ASM Disk Group "+DATA" and another ASM Disk Group "+RECO". The DATA Group should contain all data files, one copy of control file, online redo log files and spfile. The RECO Group should contain second copy of control file, temporary files, archive logs, flashback files. The DATA Group should use a redundancy level of high and the RECO Group a redundancy level of normal. OCR and voting disks are stored in internal standard Exadata diskgroups, called +DBFS_DG or +SYSTEMDG where name is Exadata version dependent and not related to Database File System.

Note: BR*Tools do not support ASM locations for backup destinations, e.g. +RECO cannot be used for disk backups created by BRBACKUP (backup_root_dir) or BRARCHIVE (archive_copy_dir). Please refer to below chapter 'SAP Database Administration with BR*Tools' for more details.

When storing more than one SAP database (for instance an SAP ERP database and an SAP BW database or an SAP ERP database with an SAP CRM database or multiple SAP ERP databases) on the Oracle Exadata Database Machine all files of each of these SAP databases should follow the above recommendation and therefore files should be stored in the DATA and RECO Group.

This recommendation is in contrast to the 3 disk group setup variants given in the Oracle Whitepaper 'Moving your SAP Database to Oracle Automatic Storage Management 11g Release 2, A Best Practices Guide'. This difference between Exadata and standard ASM setup reflects the different storage system architectures, as Exadata isn't just a plain disk storage device. On Exadata the two disk groups +DATA and +RECO are preconfigured and should be used as configured. Of course the setup might be changed toward a more detailed configuration, but only in event of additional special requirements.

For performance and throughput reasons it is recommended to only have two control files, one copy stored in the +DATA, the second copy in +RECO group, but non-multiplexed online redo log files (one member per redo log group) for each SAP database. As standard, non Exadata, SAP installations use three control files in the database, it is recommended to remove one control file from the spfile or init.ora. Standard SAP installations also use two members for each online redo log file. On the Oracle Exadata Database Machine it is therefore necessary to remove one member of each online redo log file group for each redo thread. The source database will have multiple redo threads if it was a RAC database. Three control files and multiplexed online redo log files are not needed on the Oracle Exadata Database Machine. The online redo log files are stored in the DATA Group, which already provides three way mirroring for each file at the Oracle ASM level due to the redundancy level of high, the control files are stored in DATA and RECO making 5 overall copies in total because RECO is providing two way mirroring due to normal redundancy level.

ASM compatibility attributes

Each ASM disk group has an attribute describing version compatibility of features offered by ASM. This attribute is divided in 3 sub attributes for RDBMS, ASM and ADVM, which is the ACFS volume manager. The last one isn't used here, as ACFS is not supported on Exadata. But the other two, RDBMS and ASM must be set to a minimum value of '11.2.0.2.0', which should be the current setting for preconfigured disk groups on Exadata. To verify these attribute settings, or to modify if necessary, please refer to chapter 'Oracle ASM compatibility Attributes' in the Oracle Whitepaper 'Moving your SAP Database to Oracle Automatic Storage Management 11g Release 2, A Best Practices Guide'.

Character Set Requirements for SAP Databases

New installations of SAP systems from NetWeaver 7.0 on are Unicode installations only. For an SAP Unicode installation it is required that both the character and the national character set in the database is set to UTF8. When deploying a new Oracle Exadata Database Machine for Unicode installations of SAP it is therefore mandatory to specify UTF8 for both the character and the national character set in the Exadata configuration worksheet. A working example for the Exadata configuration worksheet can be found in Appendix 1 of this paper.

So please make sure that the default database on the Oracle Exadata Database Machine is created with the SAP required UTF8 character and national character set or a new SAP database must be created on the Oracle Exadata Database Machine for instance through DBCA with a character and national character set of UTF8.

Non-Unicode SAP Installations

Existing Non-Unicode SAP installations can be used with the Oracle Exadata Database Machine. It is important for these Non-Unicode installations that the character and national character set of the migrated databases from existing systems to the Oracle Exadata Database Machine is kept the same as in the original system. The character set will either be WE8DEC or UTF8. The national character set should always be UTF8.

It is mandatory that the SAP application of such a Non-Unicode SAP installation runs on an operating system which supports the Non-Unicode runtime requirements of SAP. The Product Availability Matrix (PAM) of SAP (http://www.service.sap.com/PAM) should be checked for valid operating system support for Non-Unicode SAP installations.

In the case of Non-Unicode installations it is highly recommended to not change the hardware or operating system for the SAP layer. Only the existing database server and storage layer should be changed to the Oracle Exadata Database Machine.

Note: In the case of Non-Unicode SAP installations you cannot run the SAP Central Services on the database nodes of the Oracle Exadata Database Machine.

SAP Database Administration with BR*Tools

The installation, configuration and operation of BR*Tools on the database nodes of the Oracle Exadata Database Machine are documented in SAP OSS notes 1598594 "BR*Tools configuration for Oracle inst. under 'oracle' user" and 1627541 "BR*Tools support for Oracle ASM and Exadata" .

Main differences between common single instance or RAC installation and Exadata in a SAP environment is, Exadata is using ASM and local Oracle homes for the database. Maybe multiple databases are operated simultaneously on an Exadata system.

Since BR*Tools version 7.20 (patch 18) ASM and Exadata are supported by BR*Tools (SAP OSS note 1428529 "Corrections in BR*Tools Version 7.20"). Because no older BR*Tools version support ASM fully, which is an requirement for Exadata, 7.20 (18) is the

minimum BR*Tools version. Functions offered by BR*Tools in Exadata environments includes:

- Backup, Restore and Recovery

- Tablespace and Datafile management

- Instance management

- Database checks

- Segment management

- Statistics, etc

So all necessary functions are included. Database backups and restore must involve RMAN.

Note: BR*Tools can only write database backups to file system, 'backint' or 'RMAN-MML' only and can restore backups from there. BR*Tool log files and protocols are written to file system and can be copied via 'backint' onto tape. BR*Tools cannot perform a backup to an ASM location or restore from an ASM location. Init<SID>.sap parameters like 'backup_root_dir' and 'archive_copy_dir' cannot point to an ASM disk location, they must point towards a proper file system location. Recommended is to attach an NFS/ZFS appliance for hosting these file systems.

Because of local Oracle homes for the database, a new init<SID>.sap parameter must be set: loc_ora_homes = yes

ASM related BR*Tools parameters are explained in SAP OSS note 1598594 "BR*Tools configuration for Oracle inst. under 'oracle' user".

On Oracle Exadata Database Machine BR*Tools can be installed in 2 ways:

BR*Tools Installation on one database node

BR*Tools will be installed on one dedicated database server only, which becomes a BR*Tools administration host. All BR*Tools activities must run on this particular node.

Advantage

- Minimal installation and administration effort (only one application server, one database node)

- BR*Tools log files reside on one node only

Disadvantage

- Only one node for BR*Tools based administration. If this node is unavailable, execution of BR*Tools tasks are impossible. This setup makes BR*Tools a single point of failure in terms of database / system administration.

BR*Tools Installation on all database nodes

To enable all Exadata database nodes to run BR*Tools, the package must be installed on all database nodes.

Advantage

- BR*Tools resists on all nodes, if one node crashes, it doesn't matter

- Easier operating in case of hardware failures

Disadvantage

- More installation and administration effort (minimum one application server, all database nodes)

- BR*Tools log files reside on all nodes locally

With this configuration you can change the hostname in the RFC Destination from a fix host to a random host in RAC-Cluster for executing BR*Tools.

- Change to transaction RFC-Destinations (SM59)

- Go to TCP/IP Connections -> SAPXPG_DBDEST_<DBHOST_NAME>

- Change the hostname from "<exadata-node1-hostname>" to "<Scan-VIP>" (Scan-Listener-Address)

Please note, that this configuration needs a shared HA-NFS setup of BR*Tools configuration and log directories, to enable the tools to access even logs created remotely.

BR*Tools installation process overview

The SAP processes run in an own Unix environment, which is usually called <SID>adm. It is created during SAP installation process on SAP application server hosts. On the Exadata database nodes, the user must be created too if any kind of SAP code is executed there. Unix user-id and group should be identical for users 'oracle' and '<SID>adm' across all over the system (SAP servers and Exadata database nodes). Please double check , that OS user 'oracle' belongs to OS group 'oper'. Environment must be maintained for <SID>adm on all Exadata nodes, mainly:

DBSID   getting the local database instance_name, e.g. KCM1

dbs_ora_tnsname   set to the connect string to the general connect service offered
                               by all database instances <DBID>, e.g.
                               tnsnames.ora: KCM.world
                               EZCONNECT: //<Scan-VIP>/KCM

Furthermore mainly the environment variables ORACLE_HOME, ORACLE_SID and SAPDATA_HOME are needed to run BR*Tools.

If the .dbenv.sh or .dbenv.csh script contains any path to Oracle home files, please verify to use the 'run time path' to Oracle home <OHRDBMS>, means the symbolic link and not the version dependent directory. See SAP OSS note 1524205 "Oracle 11.2.0: Database Software Installation" for details.

The database should contain a database user called 'OPS$<SID>adm', which is usually existent, if the database was migrated from an existing SAP system. In event this user does not exist, please refer to the following SAP OSS note for details on user creation: 361641 "Creating OPS$ users on UNIX".

On all Exadata database nodes the OS user <SID>adm should be able to log on to the database using the command: sqlplus /

The installation process of the BR*Tools package itself is described in detail in SAP OSS note 1598594 "BR*Tools configuration for Oracle inst. under 'oracle' user". Because Exadata doesn't support a shared file system and all Oracle homes are local, on each Exadata database node the following directories must exist (owned by oracle:oinstall):
/oracle/<DBNAME>/saparch
/oracle/<DBNAME>/sapbackup
/oracle/<DBNAME>/sapcheck
/oracle/<DBNAME>/sapreorg
/oracle/<DBNAME>/sapprof
/oracle/<DBNAME>/saptrace

For administration reasons somebody might like to share these directories via NFS, as Exadata doesn't provide shared file systems. Mainly saptrace directory should not be shared. Stale NFS handles on log or trace files can hang up local database instance up to the whole database, which the needs to be restarted completely. To avoid this dependency, saptrace directory should not be shared via NFS.

After installation has completed, it can be tested using: brconnect -u / -f check

To enable remote administration between Exadata database nodes and administration started via SAP transactions (e.g. DB13), it is recommended to setup:

- ssh connection without password for OS user <SID>adm between Exadata database nodes

- ssh connection without password for OS user 'oracle' between Exadata database nodes

- ssh connection without password between OS users <SID>adm and 'oracle' between Exadata database nodes

- password less ssh connection from SAP application servers to Exadata database nodes is undesired. Better is to use the standalone gateway, which needs the SAP utility sapxpg on each Exadata database node

For more information, refer to SAP OSS note 1025707 "DBA Cockpit: Planning calendar and remote Oracle databases", which also includes information about the following steps.

The sapxpg utility is part of SAPEXE.SAR but can also be downloaded as a standalone package from 'service.sap.com/swdc'. Navigate as follows:
1. Support Packages and Patches
2. A-Z Index
3. K (for Kernel)
4. Choose right SAP kernel version (Exadata only supports unicode SAP kernels)
5. Select OS running on your Exadata (Linux on x86_64 64bit)
6. Click on Database independent

7. In the box below, select e.g. sapxpg_<kernel-version>.sar for download

Using <SID>adm environment on Exadata the sapxpg is installed like this:

1. Goto sap-exe directory (command: cdexe or cd /usr/sap/<SID>/SYS/exe/run)
2. SAPCAR -xvf <path to sapxpg package>
3. verify owner and group of sapxpg, must be <SID>adm:sapsys
4. start SM59 and goto 'TCP/IP connections'
5. double click on SAPXPG_DBDEST_<DBHOST_NAME>
6. hostname might be adjusted
7. verify sapxpg path, should contain absolute full path
8. save and test with 'test' button

Additional detailed information can be found in SAP OSS notes:
108777 "CCMS: Message 'SAPXPG failed for BR*Tools'"
387137 "RFC connection test for sapxpg does not work"

In addition to above installation procedure, the SAP instance profiles should contain gateway specific parameters like these:

- gw/rem_start   =   SSH_SHELL

- gw/remsh   =   /usr/bin/ssh

If the remote host is a cluster or if you want to start the BR*Tools on a specific real application cluster (RAC) node, the virtual cluster host name or the host name of the RAC node must be entered in the DBA Cockpit under "Jobs -> Back-End Configuration", that is in the dialog box "BR*Tools execution host" that can be called by using the menu "Administration -> ORACLE Settings".

Hint: After installing the first node, for a faster install process , you can copy BR*Tools and sapxpg under "/usr/sap/" to the other Exadata database nodes.

The file /oracle/<SAPSID>/sapprof/init<SAPSID>.sap will be linked to all database instances. This is to have only one init<SID>.sap to handle, if changes apply.

```
$ ln -s /oracle/<SID>/sapprof/init<SID>1.sap
/oracle/<SID>/sapprof/init<SID>2.sap

$ ln -s /oracle/<SID>/sapprof/init<SID>1.sap
/oracle/<SID>/sapprof/init<SID>3.sap

$ ln -s /oracle/<SID>/sapprof/init<SID>1.sap
/oracle/<SID>/sapprof/init<SID>4.sap
```

Lifecycle Management for SAP Databases

An Oracle Exadata Database Machine requires life cycle management at several levels of its hardware and software stack:

- Exadata Storage Server

- Database server

  ○ Oracle Database Server

  ○ Operating system and firmware

- InfiniBand switch

- Additional components

This section focuses on the Oracle Database Server and describes how to install Oracle Database Server software patches into the Grid Infrastructure Oracle Home and the RAC Oracle Home of an SAP database. For more information on the other components mentioned above, see MOS note 1262380.1 "Exadata Patching Overview and Patch Testing Guidelines".

The Oracle Database Server of an SAP database requires two bundle patches for a complete update:

- The regular Oracle Exadata Database Machine Bundle Patch (which contains patches for Database, ASM and Clusterware), also referred to as "Oracle Exadata Bundle Patch" in the following, and

- The SAP Bundle Patch for Oracle Exadata, also referred to as "SAP Exadata Bundle Patch" in the following.

Oracle tests and certifies both bundle patches for SAP databases on a regular basis and makes them available for SAP customers on the SAP Service Marketplace. You can find up-to-date release information on both bundle patches and their download locations in SAP OSS note 1591389 "Exadata 11.2.0: Patches for 11.2.0.2" and SAP OSS note 1656654 "Exadata 11.2.0: Patches for 11.2.0.3".

Since each SAP Exadata Bundle Patch requires a specific version of the Oracle Exadata Bundle Patch, you cannot use versions of the Oracle Exadata Bundle Patch for SAP databases which have not been specifically certified for that purpose.

The SAP Exadata Bundle Patches can be installed on top of regular Quarterly Full Stack Download Patches (QFSDP) for Exadata which include Quarterly Database Patches for Exadata (QDPE). The SAP Bundle Patches are based on the Exadata Bundle Patch used as QDPE.

For Oracle 11.2.0.3 initially Oracle Exadata Bundle Patch BP3 is used to build up the first Exadata SAP Bundle Patch for 11.2.0.3 whereas BP2 is the QDPE. When installing this Exadata SAP Bundle Patch BP2 will be removed if existing and BP3 will be installed which in turn includes BP2 (cumulative).
More information on QDPE can be found in Oracle MOS note 888828.1 "Database Machine and Exadata Storage Server 11g Release 2 (11.2) Supported Versions".

Installation of the OPatch and MOPatch Utilities

The installation of both bundle patches requires an up-to-date version of the OPatch utility. The installation of the SAP Exadata Bundle Patch additionally requires an up-to-date version of the MOPatch utility.

Appropriate versions of both utilities are available in the SAP Exadata Bundle Patch. See section "OPatch and MOPatch Utility Information" in the Readme document of the SAP Exadata Bundle Patch for instructions on how to extract and install these utilities.

Installation of the Oracle Exadata Bundle Patch

The Oracle Exadata Bundle Patch contains patches which must be installed in the Grid Infrastructure Oracle Home (GI Home) and patches which must be installed in the RAC Oracle Home (RAC Home) of an SAP database.

In case you have not installed an SAP system on the Exadata system yet you can use OneCommand to install the QFSDP / QDPE.

As soon as you have installed at least one SAP system on the Exadata system the recommended option to install the Oracle Exadata Bundle Patch is documented in the Readme document of the SAP Exadata Bundle Patch. Currently you should use "opatch napply" as described in the Readme section "Installing the SAP Bundle Patch".

The patchset version for GI Home and RAC Home have to be identical, but the Oracle Exadata Bundle Patch version do not have to be the same. Supported combinations can be found in SAP OSS note 1677978 "Administration of mixed SAP/Non-SAP Environments on Exadata".

After you have installed the Oracle Exadata Bundle Patch you should immediately install the SAP Exadata Bundle Patch. In particular, you should not execute the SQL statements mentioned in section "Patch Postinstallation" of the Readme document, since the catsbp.sql script provided by the SAP Exadata Bundle Patch runs also the SQL statements required by the Oracle Exadata Bundle Patch.

Installation of the SAP Bundle Patch for Oracle Exadata

You must install the SAP Exadata Bundle Patch after you have installed the Oracle Exadata Bundle Patch.

Note: The SAP Exadata Bundle Patch is not installed by the Oracle OneCommand Exadata installation utility.

The SAP Exadata Bundle Patch contains patches which must be installed in the Grid Infrastructure Oracle Home (GI Home) and patches which must be installed in the RAC Oracle Home (RAC Home) of an SAP database. Use the MOPatch utility as described in section "Installing the SAP Bundle Patch" of the Readme document of the SAP Exadata Bundle Patch to install the patches either into

- all RDBMS homes (non-rolling)

- all SAP homes (non-rolling), non-SAP homes (rolling)
- single-instance ASM installation

of the Oracle Exadata Database Machine.

Finally, follow the post-installation steps in section "Executing Post-Installation Instructions" of the Readme document of the SAP Exadata Bundle Patch to run all required SQL statements, update the database dictionary, and maintain the database initialization parameters.

Migration of SAP Databases

Although there are several possibilities to migrate an existing SAP database to the Oracle Exadata Database Machine, it is recommended to choose one of the following approaches described below as these have been successfully tested. Main task is the migration to ASM, which is described in the Oracle Whitepaper 'Moving your SAP Database to Oracle Automatic Storage Management 11g Release 2, A Best Practices Guide'. However there are some important differences between the standard ASM setup used in the white paper above and Exadata. Exadata operates databases and ASM in RAC / cluster mode, so multiple nodes could be used as a target for migration depending from migration approach being used and the setup of source database. Additionally the ASM setup differs slightly between Exadata and the setup used in the white paper.

If the source database is operated in single instance mode, it can be additionally migrated to RAC after moving it into ASM on Exadata. Enabling of multiple instances /RAC is optional, but in most cases it will happen probably. The RAC migration is described further down in an extra chapter. In all database migration approaches below, the source database could be operated in single instance mode or RAC / cluster mode. The above white paper only describes single instance databases for source and target, which is the easiest variant. Here also single instances are covered for source. If multiple RAC instances must be used during migration for performance reasons, this requires additional steps not described in this document.

Please refer to chapter 'Post Migration Tasks' for any task to be executed after migration has completed.

Migration Approach 1: Oracle-to-Oracle offline migration (O2O) ACS Service and Customer Self-Service

This way of database migration exists for many years and is being used to migrate an SAP database between different systems. The method is also described in the SAP OSS note 1508271 "Oracle to Oracle Online Migration - Triple-O".

The O2O database migration method was developed by the Oracle ACS service, to offer customers, having very large database, a fast, smooth and reliable migration method. This method offers a migration speed of more than 1 TB/h and reduces the efforts which must be spent into in configuration and testing of the migration.

O2O supports all operating systems SAP products are certified on. Because O2O is operating system independent, it can be used to perform homogenous and heterogeneous system copies. A homogenous system copy is a migration where the source and target

operating system is the same. A heterogeneous system copy is a migration where the source and target system have different operating systems. With an heterogeneous system copy you can for instance migrate an existing SAP AIX database to an Exadata Linux database.

The advantage of this method is, that you can combine the operating system change with multiple options to get most out of the migration:

- With O2O it is possible to combine a platform migration with a release upgrade. The migration method supports every combination of Unix, Windows or Linux on source and target system. So you can migrate an existing Oracle 10.2 database on HP-UX to an Exadata Solaris database.

- It is possible to upgrade directly to an higher database release. Currently with the O2O method direct database migrations are possible between different Oracle versions. So it is possible to upgrade directly from Oracle 9i to Oracle 11g by using O2O. You also do not need the most current patchset of the lower Oracle release to run the migration. A complete overview about the upgrade paths between different Oracle versions is given at the end of this chapter.

- As part of the database migration, the whole database is reorganized. This can free up a significant amount of space within tables and indexes.

- The tablespace layout can be changed to the new SAP standard or to a customer own customized one. It is also possible to move single tables and indexes to separate tablespaces or to merge them into existing or new ones. This allows you to unify the SAP landscape by using a default tablespace name like "PSAPSR3" in all SAP systems

- The SAP schema name can be changed for instance to "SAPSR3" to unify the SAP landscape.

- The number of data files and mount points can be significantly reduced, by either optimizing the tablespace layout or the size of the data files and file systems

- Tablespaces are created with LMTS and ASSM.

- The method creates the appropriate scripts to create either a database using normal files systems or a database using ASM or Exadata file systems.

- Data files will be converted from file system to Oracle ASM. The method provides you with the needed scripts to create the database and the tablespaces automatically.

- LOB or LONG data types can be converted to Secure files (11.2 only). Securefiles will also be compressed, if possible.

- You can compress the indexes on the target database by using Oracle index compression. The compression calculation is executed as described in SAP OSS note 1109743 "Use of Index Key Compression for Oracle Databases". The correct index compression for each index is determined automatically.

- You can compress tables on the target system. The compression will compress all SAP tables as recommended in SAP OSS note 1431296 "LOB conversion and table compression with BRSPACE 7.20".

The downtime needed to migrate a database with the O2O method is depending on the database size, the included database objects (SAP cluster tables, partitioned tables) and the available hardware resources(CPU, Memory, Storage, Network). Up to 1TB/hour is possible.

Although O2O was originally developed to perform database migrations, it can be an interesting method to perform a simple database upgrade, because you can implement all new database features you want into a single step:

- database upgrade directly from 9i or 10g to 11g. There is no need for a special database release on the source, which can safe additional time, because you don't have to patch the current database release before you can upgrade

- Reorganistion of the complete database to free up unused space

- Implementation of index and table compression, which will result in a 50% smaller database size

O2O Technology

- The O2O method is based on the following few steps approach:

- Prepare the source system, by defining the needed Oracle directories on the file system and load the PL/SQL into the source system. The package is only a few MB in size. To hold the scripts on the file system, only a few MB are needed (Typically less than 50 MB)

- Define the basic conditions for the migration, e.g. definition of the setup of target database, (ASM configuration, usage of table and index compression,or other database features). The PL/SQL package is then executed to generate the needed migration scripts. A typically package run will take 30 min – 1h on a SAP system.

- Creation of an empty target database, either with the provided scripts generated by the PL/SQL package or with your own scripts

- After the scripts are created, they are copied to the target system, to run the database migration itself. To run the migration process a program named "scheduler" is available, which runs all needed migration scripts in the correct order and controls also the correct execution of each script. Using this scheduler you have full control over the migration. This includes the possibility to restart failed scripts or to set it to "Done". The scheduler is written in ksh and runs on all Unix or Linux operating systems. It's also possible to run the scheduling software on a remote machine, e.g. if Windows is used on source or target.
The throughput depends on the available hardware on source and target machine. In best case you can build up the target database with more than 1 TB/ hour. Typically the throughput will be between 250 GB/h and 500 GB/h in average.
To achieve this throughput, different migration methods are used for the database tables. Based on the table size and datatype (e.g. SAP cluster tables), the best migration for the particular tables is chosen to achieve the best migration performance. Typically most of the data is transferred directly over

the network, not using a dumpfile for the migration. This saves space on the file system.
To verify the migration, source and target databases are compared on object level (based on the object name) to ensure the correctness of the migration. In a second step, In addition all tables on source and target are compared based on the number of rows. So at the end of the migration you can prove the correctness of the migration based on object and row level.

- Once the migration is completed the post SAP migration tasks can be started. The system are fully supported by SAP as mentioned in SAP OSS note 1508271 "Oracle to Oracle Online Migration - Triple-O"

- The O2O method is developed and maintained by the Oracle ACS service in Walldorf. O2O is either available as an ACS service or can also be used by customers themselves.

Migration Approach 2: Oracle-to-Oracle online migration (Triple-O) ACS Service only

If the downtime requirements cannot be fulfilled with the O2O offline method, you can use as an alternative the Triple-O method, also described in SAP OSS note 1508271 "Oracle to Oracle Online Migration - Triple-O". The estimate the downtime available for the technical database migration, you have to get the downtime for the application first. In this downtime not only the database migration must be performed, but in case of a heterogeneous system copy , user acceptance test, interface tests and post-SAP migration tasks must be performed. As a result the real available downtime for the technical database migration is much shorter than the application downtime.

So the Triple-O method was developed, to reduce the technical database migration time to a minimum. Typically the required downtime for the database migration is between 30 minutes and 1 hour, independent from the database size. On customer request the source and target database can be compared based on the row count after the migration is completed, to ensure the correctness of the online migration.

Triple-O technology

Triple-O allows are real online migration of the database.. It uses GoldenGate software for the online synchronisation of the changes and a modifies O2O version, to perform a consistent initial database load. GoldenGate reads the online redo log or the archive log of the database and extracts DDL and DML changes, recorded in the redo log of the database. GoldenGate transforms the physical changes from the redo log file into a general description, which is operating system and database release independent, and saves these changes into trail files. The volume of the trail files is typically 30% - 50% of the redo log volume. To use GoldenGate supplemental logging must be enabled on the database.

The GoldenGate trail files are sent to the target database server, either with GoldenGate, unsing a network connection, or by providing the trail files via NFS to the target system. On the target side the trails files are read by apply processes, which execute native SQL statements, generated out of the contents of the trail file on the target system. So each

DML is transformed into the appropriate insert, update or delete command. This allows to use GoldenGate for heterogeneous system copies as well.

It's possible to start or stop an online migration at any time. There is no downtime needed for the start or the stop of the migration. Furthermore the migration method doesn't need any special database patches, nor a special database patch level. Triple-O runs on any 9i, 10g or 11g version and supports DML and DDL changes. So there are no limitations (e.g. transport stop) on the usage of the SAP system during the migration. All operations can be performed quite normal. Both, R/3 and BW systems are fully supported.

Because GoldenGate is operating system independent, also heterogeneous database migrations are supported. The minimum Oracle release on source is 9.2. Direct database upgrades to 10g and 11g are possible. When using Triple-O, you can make use of all features listed for the O2O method. So the basic configuration is very similar between O2O and Triple-O. Beside the migration scripts, for Triple-O also the GoldenGate configuration scripts will be generated automatically.

The online migration is running in a very similar way, as the previously described O2O migration. A PL/SQL package is loaded in to the database, and creates all necessary script, to perform the online migration. Instead of running the migration scripts offline (SAP is down), these scripts are now executed, while SAP is up and running. To allow a consistent export of the data, Oracle flashback capabilites are used. This allows on one side a consistent export of the tables to a specific SCN (System Change Number), on the side it's possible to configure the GoldenGate processes for each table to the table specific SCN, to guarantee an error free apply. The fetch of an individal SCN for each table and the update of the GoldenGate configuration files is executed by the scheduler software, which runs the migration.

One challenge in an online migration is to handle the additional load, generated by the GoldenGate processes and the initial dataload. Very often the current hardware on the source system is outdated and already fully used by the current application load. So it's not possible to add additional load on the system, without harming the production. To allow an online migration even under these circumstances, it's possible to run a downstream capture of the GoldenGate processes. In such a configuration, the archive files from the production system are analysed and extracted on a different server. This server must not have the same OS but only the same endian byte order. So it's possible to process database archive files from HP-UX, AIX and Sun Solaris (big endian) and Linux, Windows (little endian) cross-platform. Therefore it's possible to run the GoldenGate processes on a different hardware, to unload the productive environment. To preserve the resources on the production system.

If it's even a problem to handle the initial data unload on the productive system, a shadow database can be used instead. So it's possible to run a Triple-o migration completely outside of the productive systems, which allows to perform an online migration even on old and outdated hardware.

To support also very large databases with a huge system load it's possible to configure GoldenGate in many different ways. To increase the capabilities to process a larger redo log volume, up to 34 GoldenGate processes can be defined. Each process gets a number of tables, to extract the DML and DDL changes of these tables. The PL/SQL package will perform a load distribution, based on the recorded DML changes in the database for

each table. It's possible to configure up tp 34 processes for the redo log extraction. One GoldenGate extract process is capable to handle up to 1 TB of redo within 24h.

For the apply on the target system in general more processes are needed, than to extract the redo log file. Typically for a normal SAP system 5 – 10 apply processes are needed. For large BW systems more than 20 apply processes can be needed. To allow a maximum flexibility for the GoldenGate configuration, for each extract process up to 34 apply processes can be defined. Once again, the assigned tables in each extract are assigned to the number of defined apply processes by performing a work load balancing. So with a 2 tier architecture more than 1000 apply processes would be possible.

The typical project plan for an online migration has typically four phases:

1. Prepare the migration, generate migration scripts

2. Start the GoldenGate processes to record the database changes either on the productive system or downstream on a different server. The recorded changes, listed in the trail files are sent over the network to the target machine, where they are stored in the file system

3. Run the SCN-based initial data load either from the productive server or from a shadow database. During the initial load the database changes are recorded permanently to the GoldenGate trail files. The duration of the initial load depends on the database size and the number of jobs which can be executed in parallel to unload the database. Typically the unload is completed within 48h.

4. Once the initial load is completed, the apply processes on the target machine are started. Because the apply couldn't be started as long as the initial data load is running, there exists now a time gap between the source and the target system, which must be closed to bring both systems into sync.

5. There must be now enough time to close the time gap between source and target database. The needed time is depending from the amount of changes, which have to apply, the kind of operations, which have to apply, the number of apply processes and the performance of the target system.

6. If the time gap between source and target system is closed, both systems can run in parallel up to the final switch.

Depending on the database size and the duration of the above described steps, depend on the system performance and the amount of generated change data. Typically an online migration starts 5 days before the final switch. But if needed the migration can start more than a week before, to have enough time for all needed activities.

Triple-O for datacenter migration

Very often systems must be transferred between data centre. This can be due to a data centre consolidation, or is needed when the provider is changed. In this case it's very likely that there exist only a limited network connection between the database centres, which doesn't provide the necessary band with to run the initial data load and the GoldenGate processes in parallel. To run a full database laod over network at least a 1 Gbyte connection is needed. For WAN connections typically 100 Mbit or even less is available. In this case it's still possible to run a full online migration, with the above mentioned

minimum downtime between 30 minutes and 1 hour. To achieve this, the following approach is used:

1. Prepare the migration, generate migration scripts

2. Start the GoldenGate processes to record the database changes either on the productive system or downstream on a different server. The recorded changes, listed in the trail files are sent over the network to the target machine, where they are stored in the file system. To optimize the usage of the network band with, the GoldenGate transfer can be compressed by a factor of approx. 5-7

3. Run the SCN-based initial data unload either from the productive server or from a shadow database. To save the band width the database is dumped out to a NAS server in multiple dump files. With a single NAS device you can achieve a throughput of 150 GB/h.

4. Once the initial unload is completed, the NAS device is detached from the source system and transported to the target data centre.

5. The NAS device is attached to the target system and the target database is loaded from the NAS device.

6. Once the initial load is completed, the applies are started on the target machine.

7. There must be now enough time to close the time gap between source and target database. The needed time is depending from the amount of changes, which have to apply, the kind of operations, which have to apply, the number of apply processes and the performance of the target system.

8. If the time gap between source and target system is closed, both systems can run in parallel up to the final switch.

Using this approach it's possible to migrate even database between date centres, which are connected only by a limited network connection.

Using the enhanced configuration option for GoldenGate (downstream capturing, usage of a shadow database for the initial database load), even very large databases with a very critical performance can be migrated with a minimum downtime. Because all operations are taking place online, even under difficult conditions an online migration can be performed, on the costs of a longer duration of the migration project, but without a longer downtime.

Migration Approach 3: RMAN Transportable Tablespaces

By using Transportable Tablespaces, it is possible to migrate an existing database from any UNIX or Windows platform to the Oracle Exadata Database Machine. This migration will create a new database on the target platform, the old source database is kept in place, so it is overall like a heterogeneous copy process.

This process is described in detail in the document "Moving your SAP Database to Oracle Automatic Storage Management 11g Release 2" and can be used as a step by step instruction. For a detail explanation, please refer to Oracle online documentation version 11.2, more precisely to the book 'Backup and Recovery User's Guide'.

For a first overview here are the main facts and limitations and for using Transportable Tablespaces (TTS):

- offline process, source database and tablespaces to be exported, must be set read-only during the process

- homogeneous and heterogeneous migrations possible

- supported platforms are given in database view v$transportable_platform

- RMAN must be used in this process

- target database must be created freshly before source tablespaces can be plugged in

- network connection and shared filesystem storage (NFS) is used for RMAN reading datafiles from source database

- single tablespace or a set of tablespaces can be transferred

- 'system' tablespaces, temporary and undo tablespaces cannot be transferred. This also applies to redo logs, which certainly aren't tablespaces.

- the set of transferred tablespaces must be self-contained, means no object included in the tablespace set must refer to an object not included in the tablespace set

The migration process includes the following steps:

- Check prerequisites, e.g. supported platforms
  On source select from v$transportable_platform, to verify if the source platform is supported. Target platform is 'Linux x86 64-bit' or 'Solaris Operating System (x86-64)' for Exadata.

- Identify all tablespaces to be transferred
  Select tablespace_name from dba_tablespaces or dba_tables to determine the tablespaces to be transferred. Usually all tablespaces containing objects of SAP schema (e.g. SAPSR3, OPS$<SAPSID>ADM) belong to the tablespace set. A transfer of a subset of tablespaces / data is technically possible but might corrupt SAP dictionary. So this is not supported.

- Optional: copy 'sapuser' table into a tablespace, which will be transported
  To copy a complete SAP system, SAP login data from old database must be copied into new database. If the table 'sapuser' resides in the tablespace 'system', it must be moved to a tablespace included in the tablespace set first. Otherwise it would not be transferred.

- Verify the set to be self contained
  Use PL/SQL procedure dbms_tts.transport_set_check to verify no object in the tablespace set has references to an object not included in the set. The whitepaper provides a script for executing the above procedure to verify a set of all non-system tablespaces.

- Open source database read-only
  To ensure for data consistency, the database must be opened in read-only mode.

- RMAN convert script must be created for migrating between platforms
The white paper provides for a small RMAN script, which must be executed. The script execution results in creation of the final RMAN convert script itself. The 'convert database' RMAN command includes all tablespaces in the convert script, so after script generation, all non transferred 'system'- undo tablespaces must be deleted from the final script

- Set tablespaces to be transferred read only
Before tablespace metedata can be exported and convert procedure can be started, it must be for sure, that the metadata of the tablespace cannot change. So additionally each transported tablespace must be set read-only.

- Create export of metadata describing all objects included in the tablespace set
This has two internal steps, first an export of metadata describing all non-table objects, second a metadata export of all tables. The white paper provides exact statements / examples how to proceed.

- Create parameterfile for target database
This is the first step to be executed on the Exadata. All steps before where done on the old source machine, this and all following steps are executed on the Exadata. As RMAN needs a running database instance, first a parameter file has to be created, so the new instance can be started on the Exadata. This is a single instance (non-RAC, cluster_database=false) only. A integration into CRS (Cluster manager is not needed). However it's recommended to prepare the instance with final data like instance name (SID), e.g. KCM1. Memory parameters can be adjusted later. Also migration single instance to RAC will be done later. There are several parameters needed for ASM migration (file creation), where the given example doesn't use Exadata standards. Mainly names of ASM disk groups differ on Exadata, please adjust.

- RMAN copy into ASM and convert process
After checking that the instance can be started, the final convert RMAN script is executed. All the source datafiles noted in the convert script must be accessible using the exact path from the script. So a NFS mount is needed. The RMAN output has to be kept, because new ASM filenames will be needed for import later.

- Create target database with 'system' tablespaces only on Exadata
To create a fresh empty database, there are multiple procedures possible. The white paper describes a way using plain SQL scripts, which is the simplest methode. It provides multiple short scripts, which can be copied and adjusted. The first script '1_createdb.sql' must be modified first, to adjust ASM disk group names to Exadata standards. Thereafter SAP roles 'sapconn' and 'sapdba' have to be created as described.

- Plug transported tablespaces into new database on Exadata
This needs again two import steps. First import the non-table metadata are imported as described in the white paper. All objects are created without table data. Before the second import can be started, 'sapconn' and 'sapdba' roles have to be granted. It's recommended to provide a script for the second import of table data. The import commend needs all ASM pathes of transported tablespace

files to be specified. For this reason, the RMAN output of the convert-run should be kept. The white paper describes how to assemble the needed import command.

- Some post steps
  After the second import, the tablespaces of the database on Exadata should be checked. If all tablespaces and datafiles are known, the read-only flag must be removed. Don't forget to adjust tablespace settings for imported users. Also check the white paper for referred OSS notes to be checked after migration. RMAN can be used for validation of the database. Missing is a setup of the final spfile, migration to RAC (adding more instances) and integration into CRS. This is described below.

Migration Approach 4: RMAN Duplicate Database

The RMAN 'duplicate from active database' approach is a very simple method to create a full copy of a complete database. It can be used offline or online, so the source database can be in open state and operating during copy process. That shortens overall downtime of migration drastically. However this approach is limited to specific platforms using the same byte endian format if used for a migration to Exadata. A step by step example is given in detail in the document "Moving your SAP Database to Oracle Automatic Storage Management 11g Release 2". For further detail explanation, please refer to Oracle online documentation version 11.2, more precisely to the book 'Backup and Recovery User's Guide'. Additional MOS notes are covering this topic:

- 369644.1 "Frequently Asked Questions about Restoring Or Duplicating Between Different Versions And Platforms"

- 1079563.1 "RMAN DUPLICATE/RESTORE/RECOVER Mixed Platform Support"

Generally the RMAN 'duplicate database' command is used for creation of a database copy. This copy is created by RMAN influenced by database parameter settings, which can define a new storage structure like an ASM destination. So this approach can be used for a migration from a file system based database into ASM or Exadata. The white paper above is focusing for a migration into ASM only and doesn't care of Exadata at all. Nevertheless it can be used for a step by step instruction for Exadata migration. Exadata specific topics of this approach are noted here together with a short overview.

Requirements and limits:

- Currently supported source operating systems are Solaris and Linux on x86-64 platform and Windows

- Network (TCP/IP) connection must exist between source and target hosts
  In stead of the above described Tansportable Tablespace approach, network will be used by RMAN, so no shared file system storage (like NFS) is needed

- Minimum source database version of 11.2.0.2 plus setting of compatible parameter '11.2.0.2' or higher. Both databases, source and target, are running the same Oracle RDBMS version.

- Online / offline migration
  The RMAN option 'from active database' enables an online copy process, so the source database is open and operable. Online migration reduces certainly the downtime of the migration. If the source database is in mount state during duplicate execution, a consistent database copy is created, if the source is in open state, recovery must be executed which is automatically done by RMAN.

The following gives a short overview of the steps to be executed to migrate a database into Exadata using RMAN duplicate:

- Check prerequisites, e.g. supported platforms
  Only 'Linux x86 64-bit', 'Solaris x86-64' and Windows are supported source platforms.

- Environment on source system
  Only some environment variables are needed on the source side, e.g. ORACLE_BASE, ORACLE_HOME, ORACLE_SID, PATH
  Usually this is already setup on source.

- Oracle password file on source
  Verify source database for a valid password file, if it is missing, create one.

- SQL*Net configuration on source
  This step is needed to access the source database from Exadata and vice versa. Most often the local setup is already in place, because non DB nodes like SAP application servers need to access the source database. In short, the source instance must be known by local (source) listener, otherwise it must be added to the local listener.ora file and the listener restarted. An entry (TNS alias) to access the old and the new (Exadata) database is needed in tnsnames.ora. The file sqlnet.ora must be consistent to the entries in the files tnsnames.ora and listener.ora. A new TNS alias is needed to access the new instance on Exadata.

- SQL*Net configuration on target (Exadata)
  It must be possible to access the new instance on Exadata from source machine. So regular static instance setup (listener.ora, sqlnet.ora and tnsnames.ora) must be in place. It is recommended to use final SID's, so instance setup must not be changed later. A static setup is mandatory because RMAN will shutdown and startup the instance via SQL*Net. CRS or GRID setup is not needed at this migration step, it will be done later. So here is a static single instance defined in SQL*Net. Unlike in a normal Exadata or RAC configuration, the scan-listener and scan-address should not be used for accessing this new Exadata instance. The hostname should be used, which is an exception here. It must be for sure, that the instance is still accessible even, if not started. This static SQL*Net setup should been removed after migration to Exadata has completed.

- Environment on Exadata
  On Exadata, the oraenv script can be used to provide correct environment. The script is reading the file /etc/oratab, so instead of maintaining the environment itself, a new entry should be added to the file for the new instance.

- Oracle password file on Exadata
  The new instance on Exadata should also have a password file with the same 'sys' password like the source database.

- Parameter file preparation
  Please use the respective chapter in the white paper "Moving your SAP Database to Oracle Automatic Storage Management 11g Release 2" and adapt the given example. Good start is a copy of source parameter file. Change parameters as given in the example with the following exceptions:

  - log_archive_dest_1
    Per default on Exadata, the archive logs would be created in the Fast Recovery Area (FRA) and the parameter log_archive_dest_1 would be unset. If BR*tools or better brarchive is used to save archived redo logs, the logs must be written into ASM but not into FRA. To achieve this log_archive_dest_1 should be set to 'log_archive_dest_1 = 'location=+RECO/<DBID>/oraarch' '. The path must be manually created in ASM using 'asmcmd' before starting database instance. Please see SAP OSS note 966073 - "Oracle Flash Recovery Area/Fast Recovery Area"

  - db_create_online_log_dest_*n*
    Both parameters should be kept as given in the white paper example. So the duplicated database still has two members per redo group. After migration, the second member should be deleted due to performance impacts as stated earlier in this document.

  - log_file_name_convert
    Follow the example as here applies the same like above. This parameter becomes obsolete after migration has finished.

  - control_files
    Only two copies are needed, one copy stored in +DATA, the other in +RECO.

  Verify these settings by starting up the new instance in nomount state. Then shutdown again.

- Verifying DB access and instance start up
  Simply try to connect with SQL*Plus from Exadata to source database using the new TNS alias. Then connect from SQL*Plus on the source system to the new instance on Exadata. The remote instance must be accessible even if the instance is not started using 'sys/<password>@<new-TNS-alias to Exadata>', so remote startup is possible. The target instance must be started in nomount state.

- RMAN script creation
  Use the example given in the white paper to create an RMAN script for database duplication. The number of disk channels being defined are used on the source side. As the RMAN session will be started on the Exadata DB machine, care must be taken with the naming and RMAN syntax: the 'connect target' will connect to the old database (which is the source for the copy process) and the

'connect auxilliary' specifies the new Exadata database. The same applies to the channels. Multiple target and auxiliary channels are possible.

- Cleanup ASM storage on Exadata
  If the 'duplicate database' approach is used multiple times to refresh a database on Exadata, old database files should be deleted in ASM using asmcmd before next duplicate starts.

- Running RMAN duplicate
  Next step is to execute above RMAN script. Recommended is to log the output into a file.

Post duplication steps

After migration using Transportable Tablespaces or Duplicate approach has finished, the result should be checked carefully on Exadata. First use views v$controlfile, v$datafile and v$logfile to verify new file location in ASM. The second redo log members should be deleted as stated above. Verify deletion process in asmcmd. The RMAN command 'validate database' should be executed to verify database physical structure and data blocks for corruption.

Additionally the pfile must be maintained to erase parameters only needed for duplication or migration. So parameter entry '*.db_create_online_log_dest_2='+RECO' ' gets obsolete after removal of the second log members. Here parameters for additional instances could be included in the file, which saves some steps in the following RAC migration. At the end, an spfile, stored in ASM, should be created. It is recommended to exchange the pfile with a new version, only containing the parameter 'spfile' pointing to the new spfile in ASM. This file must exist in the RDBMS home of all Exadata DB nodes.

On the Exadata migration machine, the entry in /etc/oratab can be deleted, as it will be automatically recreated in the following RAC migration.

Migration from Single Instance to RAC

After migrating the database into ASM on the first Exadata node, the database is still in single instance mode so the remaining instances on the other Exadata nodes must be enabled. Usually each Exadata nodes operates one instance of the SAP databae, but this is not a must, also a subset of Exadata nodes can be used for hosting one specific database. But generally, after migrating onto Exadata, all further nodes get a RAC database instance.

On the database side only two changes are required to enable multiple RAC instances from a single instance database:

- Every RAC instance needs an own undo tablespace, so for every newly created instance a new undo tablespace must be created like this (for 2nd instance):
  create undo tablespace PSAPUNDO_002 datafile '+DATA' ;

  Note: the first (old single) instance should use undo tablespace PSAPUNDO, any further instance gets the undo tablespace name PSAPUNDO_00<#> where <#> is the instance or thread number. Please keep this naming scheme for avoiding future upgrade issues due to SAP tools.

- Each RAC instance needs an own set of online redo logs called 'redo thread'. The first (old single) instance get thread number 1, so for all newly created instances a new redo thread is created:
  alter database add logfile thread <n> group <m> ('+DATA…') size 200M;

  Here <n> is the redo thread number (usually equal to instance number) and <m> is the group number (unique in database). On Exadata, we keep only one member per redo log group in +DATA ASM group. Because +DATA is a 'high' redundant disk group, ASM internally keeps 3 copies of redo logs.
  After this don't forget to enable the new redo thread:
  alter database enable thread <n>;

Before starting any new RAC database instance, some RAC required parameter settings must be applied. Please note the different values for <SID> or '*' at the beginning of parameter name, they tell you the scope of parameter, so we differ between parameters valid for all instances or for a specific particular instance.

Table 1 RAC specific database parameters

Parameter Name	Value	Comment
*.cluster_database	true	database is a RAC database
<SID>.instance_name	<DBID><seq-nr>	Unique name of instance
<SID>.instance_number	<seq-nr>	Unique number of instance
<SID>.local_listener	Listener entry in tnsnames.ora	only if listener port ne 1521
<SID>.remote_listener	//<Scan-VIP>:<port>	VIP of SCAN listener
<SID>.undo_tablespace	<undo-tablespace_name>	Name of undo tablespace
<SID>.thread	<seq-nr >	Number of redo thread
<SID>.service_names	<DBID>,<instance_name>	Default instance services

For ease of maintenance and consistency <SID> of the instance-specific parameters be equal to the respective instance_name.

In the user environment only one parameter needs special attention:

- ORACLE_SID – Is equal to the <SID> of the parameters.

To be able to start an instance from sqlplus create a file init<SID>.ora for every local instance in the local RDBMS home containing:

spfile='<path/name to the spfiles>

Example: spfile='+DATA/<DBID>/spfile<DBID>.ora'
Certainly, the file must already exist …

Then the database and all it's instances has to be defined as a CRS resource:

```
srvctl add database –d <DBID> -o <ORACLE_HOME> -p <path/name of spfile>

srvctl add instance –d <DBID> -i <SID> -n <Exadata node>
```

Now all RAC database instances can be started. To maintain the file /etc/oratab on all the nodes, please refer to the SAP OSS note 1554661 "Configuration of environment for 'oracle' user".

Post Migration Tasks

The following is a list of tasks to be executed after migration itself has finished. Not all tasks apply always, please check which tasks are needed for your system.

- Create ASM aliases for spfile and control files

- O2O post activities

- Check user OPS$oracle

- Additional database services

- Customize SAP Profiles

- Customize tnsnames.ora on SAP Applicationsserver

- R3trans Test

- Check/Implement SAP OSS note 743555 "Oracle Sequence for RAC"

Create ASM aliases for spfile and control files

As described in SAP OSS note 1598594 "Configuration of environment for 'oracle' user" both control file copies and the spfile should get a standard alias in ASM pointing to the real files. This is needed to enable BR*Tools to refer to these files. The aliases must look like this:

```
+DATA/<DBID>/spfile<DBID>.ora
+DATA/<DBID>/cntrl<DBID>.dbf
+RECO/<DBID>/cntrl<DBID>.dbf
```

Use asmcmd utility to create them

```
ASMCMD> mkalias +DATA/<DBID>/PARAMETERFILE/spfile.<nn>.<mm>
+DATA/<DBID>/spfile<DBID>.ora
ASMCMD> mkalias +DATA/<DBID>/CONTROLFILE/Current.<nn>.<mm>
+DATA/<DBID>/cntrl<DBID>.dbf
ASMCMD> mkalias +RECO/<DBID>/CONTROLFILE/Current.<nn>.<mm>
+RECO/<DBID>/cntrl<DBID>.dbf
```

The spfile aliases must be used in local init<DBID>.ora in $ORACLE_HOME/dbs, so it should look like

```
spfile = '+DATA/<DBID>/spfile<DBID>.ora'
```

The two control file aliases must be used in control_file database parameter stored in spfile. The parameter should refer to the aliases and not the 'real' ASM filename

```
control_files = '+DATA/<DBID>/cntrl<DBID>.dbf',
'+RECO/<DBID>/cntrl<DBID>.dbf'
```

O2O post activities

If you migrate your database with Toolset O2O public synonyms will not be migrated. After migration execute BR*Tools to create standard SAP public synonyms:

```
brconnect -c -u / -f crsyn -o <SAP Schema for example SAPSR3>
```

Check OPS$oracle user

If the user OPS$ORACLE does not exist, create it in the database as follows:

```
SQL> connect / as sysdba

SQL> create user ops$oracle identified externally;

SQL> grant sapdba to ops$oracle;
```

Oracle User "system" should have privilege "sysoper" to operate correctly with BR*Tools ("grant sysoper to system").

Additional database services

After migration of data and migration to RAC the database services should be customized. This allows the SAP Application Server to connect to the database without configuring tnsnames.ora on Exadata.

When migration and RAC setup has completed, all instances offer two static services provided by database parameter 'service_names', which are named like <DBID> and <instance_name>. These services are static and always offered. The <DBID> service is e.g. used by R3trans or TP, while the second is for instance administration purposes.

Additionally to the static services, each SAP Application Server Instance gets its own database service on Exadata:

```
srvctl add service -d <DBID> -s <Service_name> -r <SID1> -a <SID2>
```

Example for database ERP with RAC SIDs ERP1 and ERP2:

SAP Applicationsservice: DVEBMGS01, D00

Former central instance DVEBMGS01 located to Exadata node 1, application server instance D00 located to Exadata node 2:

```
$ srvctl add service -d ERP -s ERP_DVEBMGS01 -r ERP1 -a ERP2
$ srvctl add service -d ERP -s ERP_D00 -r ERP2 -a ERP1
```

Database services will be listed under Oracle Parameter "SERVICE_NAMES"

```
[oracle@exa1db01 sapprof]$ sqlplus / as sysdba
SQL> show parameter service
NAME                         TYPE      VALUE
---------------------------- --------- ----------------------------
service_names                string    ERP_SCS00, ERP_D00, ERP
```

Usefull Commands for database services

- srvctl disable database -d ERP; srvctl enable database -d ERP

- srvctl config database -d ERP

- srvctl stop database -d ERP; srvctl start database -d ERP

- srvctl status database -d ERP
  Instance ERP1 is running on node exa1db01
  Instance ERP2 is running on node exa1db02

- srvctl stop listener; srvctl start listener; srvctl status listener

- srvctl stop scan_listener; srvctl start scan_listener; srvctl status scan_listener

- srvctl status service -d ERP
  Service ERP1 is running on instance(s) ERP1
  Service ERP2 is running on instance(s) ERP2
  Service ERP_ASCS02 is running on instance(s) ERP2
  Service ERP_D00 is running on instance(s) ERP2
  Service ERP_DVEBMGS01 is running on instance(s) ERP1

- srvctl remove service -d <DBSID> -s <Service>

Customize SAP Profiles

After the database is migrated to Exadata the SAP Profiles must be adjusted to connect to the new RAC-Database on Exadata. Mainly SAPDBHOST and Java-DB-Host must be adjusted as shown below.

- SAP Default Profile

  - SAPDBHOST = <Exadata-VIP of node 1>

  - j2ee/dbhost = <Exadata-VIP of node 1>

If tnsnames.ora files are still be used and all SAP instances are already using their own TNS connect descriptor, start and instance profiles must not be changed. In this case only the connect descriptors in the tnsnames.ora file must be changed as shown in the next chapter. If tnsnames.ora files are used further but all SAP instances were using the same TNS connect descriptor, below given variables dbs/ora/tnsname and dbs_ora_tnsname must be adjusted in a way that each SAP instance gets its own TNS connect descriptor.

Recommended is to use the new EZCONNECT feature, which needs the profiles to be adjusted as follows.

- SAP Start Profile

  - EZCONNECT = //<Scan-VIP>/<TNS-service-name>

  - SETENV_03 = dbs_ora_tnsname=$(EZCONNECT)

- SAP Instance Profile

  - dbs/ora/tnsname = //<Scan-VIP>/<TNS-service-name>

Example for 2 SAP instances DVEBMGS01 and D03 of SAP system ERP:

START_ DVEBMGS01_<hostname>

>   EZCONNECT=//<Scan-VIP>/ERP_DVEBMGS01.WOLRD
>   SETENV_03 = dbs_ora_tnsname=$(EZCONNECT)

START_ D03_<hostname>

>   EZCONNECT=//<Scan-VIP>/ERP_D03.WOLRD
>   SETENV_03 = dbs_ora_tnsname=$(EZCONNECT)

ERP_DVEBMGS01_<hostname>

>   dbs/ora/tnsname = //<Scan-VIP>/ERP_DVEBMGS01.WOLRD

ERP_D03_<hostname>

>   dbs/ora/tnsname = //<Scan-VIP>/ERP_D03.WOLRD

Customize tnsnames.ora on SAP Applicationsserver

After migration to Exadata all TNS aliases used must now point to the new database to connect all SAP Application Servers with the new database. Generally all HOST parameters must use the Scan-VIP, the only exception is a connect descriptor used by BR*Tools (brspace) to remotely start or stop a dedicated RDBMS instance, these TNS decriptors are using the node-VIP in HOST parameter.

If tnsnames.ora files must be used further and all SAP instances are currently using one shared TNS alias, please create one TNS alias per SAP instance. A detailed description is given in the Oracle white paper "Configuration of SAP NetWeaver for Oracle Grid Infrastructure 11.2.0.2 and Oracle Real Application Clusters 11g Release 2: A Best Practices Guide".

R3trans Test

Use R3trans to test database connectivity. If problems occur:

- Check environment variable "dbs_ora_tnsname = <DBSID>" is set. This should be add to the environment of the user <sid>adm (.dbenv.sh and .dbenv.csh).

- Optional check/install SAP-OSS note 50088 "Creating OPS$ users on Windows NT/Oracle"

- New password generation: "orapwd file=$ORACLE_HOME/dbs/orapw password=<password>"_for_Oracle user sys.

Check/Implement SAP Note 743555 – Oracle Sequence for RAC

When you migrate your database from single instance to RAC implement SAP OSS note 743555 "Oracle Sequence for RAC" to recreate Oracle sequence DDLOG_SEQ for RAC with 2 nodes.

Shared Filesystems in SAP Environments

In an SAP environment it is common that all SAP Application Servers have access to a shared filesystem (/sapmnt, /usr/sap/trans, ...) which store the SAP kernels, profiles, trace files and provide the global SAP transport directory. In typical SAP installations such a shared filesystem is implemented using a NAS appliance, a cluster filesystem or through an NFS exported filesystem from the database server. For high availability reasons a cluster filesystem is being used or the source of the NFS location is protected by special configurations such as HA-NFS to not be a single point of failure in an SAP environment.

If you already have an existing shared filesystem solution in your SAP environment not using an NFS exported filesystem from the database server, it is recommended to continue to use this solution when moving to the Oracle Exadata Database Machine.

If the shared filesystem is NFS exported from the database server in your existing environment it is necessary to implement the shared filesystem on a different system other than the Oracle Exadata Database Machine as the Oracle Exadata Database Machine does not offer any HA-NFS or cluster filesystem capability. ACFS is currently not supported by the Exadata Linux kernels and DBFS, which would be available on Exadata, cannot be used in SAP environments. The preferred solution is to use a separate Sun ZFS Storage Appliance. Other NAS appliances, HA-NFS or cluster filesystem solutions can be used as well. The Sun ZFS Storage Appliance can also be used for very fast backups from the Oracle Exadata Database Machine by directly connecting the ZFS Storage Appliance to the Oracle Exadata InfiniBand fabric.

Protection of SAP Central Services

To create a highly available SAP system, additionally to the database the Central Services (SCS, ASCS) and the Enqueue Replication Server (ERS) must run protected by a cluster solution. Almost every high availability software like IBM PowerHA, HP Serviceguard, Veritas Cluster Server, Oracle Solaris Cluster or Oracle Clusterware provide additional services to protect these critical SAP central services. In typical SAP environments this high availability software is running either on the clustered database server or outside the database server on a separate cluster of servers.

If a separate cluster other than the database cluster is already used for the SAP Central Services then it is recommended to continue to use this separate cluster for the SAP Central Services when deploying the Oracle Exadata Database Machine.

If the SAP Central Services are not cluster protected, it is highly recommended to put them under cluster control, whereas we differ here between 'unicode' and 'non-unicode' SAP systems.

Non-unicode:

Should the SAP Central Services run on the clustered database server in your existing environment then you should consider to install two additional x86_64 systems running Oracle Solaris and Oracle Solaris Cluster on these two x86_64 systems to protect the SAP Central Services.

Note: Oracle Solaris Cluster on x86_64 hardware cannot be used for the SAP Central Services for Non-Unicode SAP installations. As an alternative for Non-Unicode SAP installations you should built an Oracle Solaris Cluster using SPARC hardware.

For unicode SAP systems:

Another alternative for Unicode-only SAP installations is to use the Oracle Clusterware running on the database nodes of the Oracle Exadata Database Machine along with the Oracle Clusterware utility SAPCTL to protect the SAP Central Services . The next chapter describes in detail all the necessary steps to install the SAP Central Services on the database nodes of the Oracle Exadata Database Machine and how to protect them through Oracle Clusterware and SAPCTL.

Note: Please be aware that any Exadata Storage Software change (patch, patch bundle or upgrade) may affect the configuration and the operation of the SAP Central Services on the database nodes of the Oracle Exadata Database Machine. So please check the configuration and correct operation of the SAP Central Services on the database nodes after an Exadata Storage Software change was applied.

Note: Please keep in mind, that although sapctl supports all SAP instance types, only SAP instance types ASCS, SCS and ERS are supported to be run on Exadata.

Installation Procedure for SAP Central Services

The SAP Central Services for ABAP and JAVA are typically provided by SAP instance type ASCS for ABAP and SCS for JAVA. The installation of these instances on Oracle Exadata Database Machine must be performed using SAP installation tool SAPINST. For a High-Availability Setup including Enqueue Replication Service (SAP instance type ERS), the initial installation must be performed on every database node on the Oracle Exadata Database Machine. Installation of instance type ASCS and SCS is fully supported by SAPINST, whereas the configuration and setup of the Enqueue Replication Service must be done manually. The following sections specify the steps required for the initial installation of SAP Central Services on Oracle Exadata Database Machine.

Installation Preparation

Please check the minimum requirements for your specific SAP software version or system type regarding OS parameters, user limits, etc. Consult SAP documentation for the recommended values and check if requirements are met on all the database nodes. If necessary, adjust at least to the minimum required value.

Download JCE (Java Cryptography Extension) policy file jce_policy-1_4_2.zip to all database nodes. Note that exactly this version is required during installation.

Assign a virtual hostname for the ASCS or the SCS instance. This virtual hostname will be used for the network name resolution and represents the IP address of the Oracle Clusterware VIP which will be used by SAPCTL to provide failover protection for the SAP Central Services.

We will refer to these virtual hostnames as **xsapdb_abap** and **xsapdb_java** throughout the following sections.

Add the virtual hostnames to the /etc/hosts file on all database nodes.

```
[root@xsapdb01] # vi /etc/hosts

  . . .

10.165.110.180  xsapdb_abap     xsapdb_abap.de.oracle.com

10.165.110.181  xsapdb_java     xsapdb_java.de.oracle.com
```

As user root, change to the directory containing the SAP Installation Master CD/DVD.

Add the variable JCE_POLICY_ZIP and SAPINST_USE_HOSTNAME to the environment.

For the installation of the SAP Central Services Instance for ABAP set

```
[root@xsapdb01] # export JCE_POLICY_ZIP=/<full-path-to>/jce_policy-1_4_2.zip

[root@xsapdb01] # export SAPINST_USE_HOSTNAME=xsapdb_abap
```

For the installation of the SAP Central Services Instance for JAVA set

```
[root@xsapdb01] # export JCE_POLICY_ZIP=/<full-path-to>/jce_policy-1_4_2.zip

[root@xsapdb01] # export SAPINST_USE_HOSTNAME=xsapdb_java
```

On the Oracle Exadata Database Machine the X11 libraries for a graphical user interface are not installed. You must use another host providing a graphical user interface for software installation with SAPINST. Export the DISPLAY variable to the host providing the GUI.

[root@xsapdb01] # export DISPLAY=<host_with_gui>:1

Now start the SAP software installation tool.

[root@xsapdb01] # ./sapinst

Follow the installation steps.

Use identical user-id and group-id for user <sid>adm on all database nodes.

The instance number for ASCS instances must be identical on all database nodes.

The instance number for SCS instances must be identical on all database nodes.

Preferable, also the password for SAP System Administrator is identical on all nodes.

Complete the installation on all database nodes.

Setup of SAP Enqueue Replication Service

The configuration task to setup an SAP Enqueue replication server instance ERS for either SAP ASCS or SAP SCS is not supported by SAPINST and must be done manually.

Check the SAP documentation for further guidance. In a nutshell we will explain the necessary steps here to give a overview on the tasks to do.

Remember that these steps must be executed on all database nodes of the Oracle Exadata Database Machine.

As user **<sid>adm**, create directory **/usr/sap/<SID>/ERS<NR>** with sub-directories **data**, **exe**, **log**, **sec** and **work**.

In the profile directory /usr/sap/<SID>/SYS/profile create the Instance and START profile for the ERS instance. Check the SAP documentation of the SAP software release installed to determine what type of profile is required to start up the instance. This may differ depending on the SAP product or version used.

The following example shows a working configuration for the START and Instance profile which can be used for reference purpose.

START profile for ABAP replication service

```
SAPSYSTEMNAME = KCM
SAPSYSTEM = 02
INSTANCE_NAME = ERS02
DIR_CT_RUN = $(DIR_EXE_ROOT)/run
DIR_EXECUTABLE = $(DIR_INSTANCE)/exe
SAPLOCALHOST = xsapdb_abap
DIR_PROFILE = $(DIR_INSTALL)/profile
_PF = $(DIR_PROFILE)/KCM_ERS02_xsapdb_abap
SETENV_00 = LD_LIBRARY_PATH=$(DIR_LIBRARY):%(LD_LIBRARY_PATH)
SETENV_01 = SHLIB_PATH=$(DIR_LIBRARY):%(SHLIB_PATH)
SETENV_02 = LIBPATH=$(DIR_LIBRARY):%(LIBPATH)
#-------------------------------------------------------------------
# Copy SAP Executables
```

```
#---------------------------------------------------------------------
_CPARG0 = list:$(DIR_CT_RUN)/scs.lst
Execute_00 = immediate $(DIR_CT_RUN)/sapcpe$(FT_EXE) pf=$(_PF) $(_CPARG0)
#---------------------------------------------------------------------
# Start SAP locking service replication
#---------------------------------------------------------------------
_ERS = enr.sap$(SAPSYSTEMNAME)_$(INSTANCE_NAME)
Execute_01 = local rm -f $(_ERS)
Execute_02 = local ln -s -f $(DIR_EXECUTABLE)/enrepserver$(FT_EXE) $(_ERS)
Start_Program_01 = local $(_ERS) pf=$(_PF)
```

Instance profile for ABAP replication instance

```
SAPSYSTEMNAME = KCM
SAPSYSTEM = 02
INSTANCE_NAME = ERS02
DIR_CT_RUN = $(DIR_EXE_ROOT)/run
DIR_EXECUTABLE = $(DIR_INSTANCE)/exe
SAPLOCALHOST = xsapdb_abap
#---------------------------------------------------------------------
# SAP Message Server parameters are set in the DEFAULT.PFL
#---------------------------------------------------------------------
ms/standalone = 1
ms/server_port_0 = PROT=HTTP,PORT=81$$
#---------------------------------------------------------------------
# SAP Enqueue Server
#---------------------------------------------------------------------
enque/table_size = 4096
rdisp/myname=ERS02_xsapdb_abap
rdisp/enqname = $(rdisp/myname)
enque/snapshot_pck_ids = 100
#---------------------------------------------------------------------
# SAPCTL HA Impl. with SAP Enq. Replication
#---------------------------------------------------------------------
rdisp/mshost = xsapdb_abap
enque/process_location = local
enque/server/internal_replication = true
enque/server/replication = true
enque/enrep/keepalive_count = 1
enque/server/threadcount = 1
```

Edit the file /usr/sap/sapservices. Add an entry for every replication server instance.

Example from a working configuration for reference purpose:

```
LD_LIBRARY_PATH=/usr/sap/KCM/ERS02/exe:$LD_LIBRARY_PATH;
export LD_LIBRARY_PATH;
/usr/sap/KCM/ERS02/exe/sapstartsrv
pf=/usr/sap/KCM/SYS/profile/START_ERS02_xsapdb_abap -D -u
kcmadm
```

Examples of the configuration files for the JAVA Central Services can be found in the SAPCTL white paper ("Providing High Availability for SAP Resources with Oracle Clusterware 11g Release 2").

Please check also all available SAP documentation and support notes on additional configuration tasks which are required for an SAP HA installation with failover capabilities.

Configuration of SAPCTL

Please refer to the latest version of the SAPCTL white paper stored on the SAP Community Network (SCN) under http://scn.sap.com/community/oracle . Follow the instructions on installation and configuration described in the collateral documentation.

Few additional tasks on the Oracle Exadata Database Machine for running SAPCTL must be executed.

After the installation and setup of SAPCTL on the first database node, copy all files and directories from /usr/sap/sapctl to the same directory on all remaining nodes. Create a symbolic link to crsctl in Grid infrastructure home directory in /usr/sap/sapctl/bin on all nodes:

[root@xsapdb01] # dcli -g <group> ln -s /u01/app/11.2.0/grid/bin/crsctl /usr/sap/sapctl/bin/crsctl

We recommend to copy the perl subdirectory from the GRID software installation recursively to directory /usr/sap/sapctl and change the ownership to <sid>adm:sapsys to avoid access permission problems.

[root@xsapdb01] # dcli -g <group> cp -r /u01/app/11.2.0/grid/perl /usr/sap/<SID>/sapctl

[root@xsapdb01] # dcli -g <group> chown -R <sid>adm:sapsys /usr/sap/<SID>/sapctl

Set variable PERL_HOME in file sapctl to point to this directory.

Note that there is no need to duplicate the START and instance profiles as described in the SAPCTL white paper if the installation of SAP Central Services for ABAP or JAVA as well as the setup of SAP Enqueue ERS replication services was performed using virtual hostnames as described in the section "Installation Procedure for SAP Central Services".

Appendix 1:

SAP Oracle_Home Naming Convention

The correct installation and operation of any SAP utility such as SAPInst or BR*Tools on the database nodes of the Oracle Exadata Database Machine requires some preparation for the correct setting of the Oracle_Home environment variable in the SAP environment. The SAP environment requires the Oracle_Home environment variable to be set to /oracle/<Database name>/<release>.

Therefore the following directories and symbolic link (according to the values listed in the configuration sheet below) must be manually created on each database node by the OS user who owns the Oracle software (In the configuration sheet below the name oracle is being used for the OS user):

root> mkdir /oracle

root> chown oracle:oinstall /oracle

root> su - oracle

oracle> mkdir -p /oracle/X11

oracle> ln -s /u01/app/oracle/product/11.2.0/dbhome_1 /oracle/X11/112

Default Oracle Environment Settings

The following configuration sheet for the Oracle Exadata Database Machine is an example for a successful SAP installation with Oracle Exadata. The sheet below shows in red the values required by SAP.

Note: Please be aware that for SAP installations only Standard OS Authentication is supported therefore Table 2 is not included in this paper.

Table 2 lists the default settings when selecting Standard OS Authentication for the OS owner used during installation to create the Oracle software environment. These default settings are in addition to the information in the configuration worksheets.

Table 2 Oracle Environment Default Settings when Using Standard OS Authentication

Oracle Database Item	Default Setting
Oracle Inventory group name	oinstall
Oracle Inventory group identifier	1001
DBA group name	dba
DBA group identifier	1002
Oracle software owner user name	oracle
Oracle software owner user identifier	1000

Oracle software owner default password	welcome
Oracle base directory (ORACLE_BASE)	/u01/app/oracle
Oracle inventory directory	/u01/app/oraInventory
Grid infrastructure home directory	/u01/app/11.2.0/grid
Database name	X11 (must be three characters)
Database character set	UTF8 (WE8DEC for Non-Unicode SAP)
Database national character set	UTF8
Database block size	8192
ASM disk groups	DATA for the default data file location RECO for the fast recovery area Note: Default DATA and RECO disk group size depends on the type of system, the type of disk drives, and the type of backup.
Exadata Smart Flash Cache	All flash disks are configured as flash cache
Starting IP address for InfiniBand private network	192.168.10.1
Subnet mask for InfiniBand network	255.255.252.0

In the general configuration worksheet (Table 2) you should always specific a workload type of OLTP for any SAP database. Even for an SAP BW database you should use the OLTP workload type OLTP as SAP BW uses serial DML (Insert, Update, Delete) and very few full table scans against the database. Most of the data access in an SAP BW database occur through indexes (Bitmap and B-trees).

Additional Information

- Exadata V2 Starter Kit (Doc ID 1244344.1)

- SAP Note 1431798 "Oracle Performance Rel. 11"

- SAP note 1590515 "SAP applications with Oracle Exadata Database Machine"

- SAP note 1554661 "Configuration of environment for 'oracle' user"

- SAP note 1677978 "Administration of mixed SAP/Non-SAP Environments on Exadata"

- SAP note 598594 "BR*Tools configuration for Oracle inst. under 'oracle' user"

- SAP Note1627541 "BR*Tools support for Oracle ASM and Exadata"

- SAP note 1428529 "ASM and Exadata are supported by BR*Tools"

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

- Oracle ASM ("SAP Databases on Oracle Automatic Storage Management 11g Release 2: Configuration Guidelines for Unix and Linux Platforms")

- Oracle Linux and SAPCTL ("Providing High Availability for SAP Resources with Oracle Clusterware 11g Release 2")

- Oracle Whitepaper 'Moving your SAP Database to Oracle Automatic Storage Management 11g Release 2, A Best Practices Guide'.

All these white papers are stored on the SAP Community Network (SCN) under http://scn.sap.com/community/oracle . The SAP Notes are available through the SAP Support Portal for authorized users.

ORACLE®

How to use SAP NetWeaver with
the Oracle Exadata Database
Machine, A Best Practices Guide
September 2013
Authors: Tanja Albrecht, Hanno
Bresch, Kurt Brög, Stephan
Bühne, Dirk Hering, Jan Klokkers,
Christoph Kurucz, Dieter Orth,
Martin Sautter, Jens Schmidt

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

Oracle Corporation
World Headquarters
500 Oracle Parkway

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

1212