

An Oracle White Paper
January 2013

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

Preface.....	3
Naming Conventions and Terms	4
Oracle ASM Disk Group Configuration	7
Oracle ASM Redundancy	7
Oracle ASM Disk Group Naming Conventions.....	8
Variant 1 – small to medium data volumes, restore time not critical	8
Variant 2 – medium to large data volumes, restore time important	9
Variant 3 – very large data and data change volumes, restore time crucial	10
Oracle ASM Compatibility Attributes	11
Prerequisites	14
Unix/Linux platforms	14
Primary group of the 'oracle' executable	14
Migrating an Active Database from File System to Oracle ASM without Changing the OS platform	14
Configuration of the Source and Target system	15
Setup the source system	15
Setup the target system.....	18
Checking your configuration	21
Checking network configuration from the source host	21
Prepare target instance	22
Create a “pfile” on the target system	22
Start the target instance into “nomount” state	24
RMAN script for "duplicate from active database"	24
Copying the database.....	25
Run "RMAN duplicate from active database"	25
Post Duplication Tasks	33
Checks.....	33

Server Parameter File	36
Register the database to the GRID	38
Start the database	38
Offline Database Migration from File System to Oracle ASM using Transportable Tablespaces	39
Steps on the source platform	40
Supported platforms for TTS	40
Identify all tablespaces and datafiles that need to be migrated	41
Change default tablespace of user SAPSR3 to 'SYSTEM'	41
Move table 'SAPUSER' to a transportable tablespace	41
Check if tablespace set is self contained	42
Create RMAN script for conversion of datafiles on the target platform	43
Export metadata	46
Steps on the target platform	49
New pfile init<DBSID>.ora	49
Convert datafiles to Oracle ASM	50
Create a new database	52
create database scripts	52
Create Database Roles 'SAPCONN' and 'SAPDBA'	54
Import all metadata (except tables of schema SAPSR3)	54
Grant 'SAPCONN' and 'SAPDBA' roles	55
Import tablespace object definitions	55
Check tablespaces	56
Switch tablespaces into read/write mode	56
Set default tablespace and temporary tablespace	57
Application of SAP notes	58
Checks, spfile, GRID registration	58

Preface

Oracle Automatic Storage Management (Oracle ASM) was introduced by Oracle when Oracle Database 10g was released. Since then Oracle ASM has proven to be one of the fastest and most reliable storage platforms to run the Oracle Database.

Oracle Automatic Storage Management 11g Release 2 has two major enhancements that are important for running SAP:

- Oracle Cluster Repository (OCR) and voting files can be stored on Oracle ASM
- Oracle RDBMS Home can be stored on Oracle's new cluster file system ACFS

This makes Oracle Automatic Storage Management 11g Release 2 the preferred storage platform for SAP running on Oracle Real Application Clusters (RAC) as well as for SAP systems running on a single instance Oracle Database.

This document gives recommendations on how to configure your Oracle ASM disk groups and describes how to migrate existing databases from file system to Oracle ASM using Oracle Recovery Manager (RMAN). The following most common migration scenarios are covered:

- Migration of an active database from file system to Oracle ASM without changing the OS platform called "RMAN duplicate from active database".
- Offline migration of a database from file system to Oracle ASM including an OS platform change using "RMAN Convert" in conjunction with "Transportable Tablespaces".

Although the focus of the examples in this document is on Unix/Linux it also covers MS Windows specific steps and most of the examples need only minor modifications.

The steps for migration from Oracle single instance databases to Oracle RAC databases on Oracle ASM are not described in this document.

You should be familiar with Oracle Clusterware 11g Release 2 and Oracle ASM. Both software components should be already installed properly according to SAP recommendations described in the according Oracle ASM notes and papers.

Customers migrating their databases to Oracle Exadata or Oracle Database Appliance may have to follow additional rules not mentioned in this document. An Oracle Exadata specific whitepaper is available on SDN. An Oracle Database Appliance specific whitepaper is planned to be released when SAP has certified Oracle Database Appliance.

Naming Conventions and Terms

The following naming conventions and terms are used in this document. For Oracle Releases (Oracle GRID Infrastructure and Oracle Database) greater than 11.2.0.2 the version numbers in names of subdirectories and environment variables must be replaced with the appropriate version numbers of the software you are going to use. E.g. For Oracle RDBMS Software 11.2.0.3 you have to use '11203' instead of '11202'.

Name or Synonym	Description / Purpose
Oracle ASM	Oracle Automatic Storage Management is basically a volume manager for all Oracle Database related files and ACFS Volumes (Oracle ASM Cluster File System Volumes).
Oracle ASM Disk	<p>An Oracle ASM disk is usually a physical disk with a logical partition on it.</p> <p>On Linux/Unix platforms the logical partition spans the whole physical disk.</p> <p>On Windows the logical partition is created as an extended partition with a logical volume in it.</p>
Oracle ASM Disk Group	A number of Oracle ASM disks form an Oracle ASM disk group. An Oracle ASM disk group has 1 or more “failure groups”. An extended partition can span part of a disk or a whole disk. Logical volumes are created inside an extended partition.
Failure Group	A named group to which an Oracle ASM disk belongs to. Oracle ASM guarantees not to store the same data twice within a failure group. Redundancy is achieved by distributing the same data to multiple failure groups.
OCR	Oracle Cluster Repository

RMAN	Oracle Recovery Manager
ACFS	Oracle ASM Cluster File System is a cluster file system that uses Oracle ASM as the underlying volume manager infrastructure.
<DBNAME>	Specifies the name of the database (e.g. C11). <DBNAME> is derived from the Oracle initialization parameter 'db_name'. If 'db_unique_name' is set then <DBNAME> is derived from 'db_unique_name'.
<DBSID>	Specifies the name of the database instance. For single instance databases <DBSID> usually equals <DBNAME> e.g. C11. With RAC the number of the instance is appended e.g. C111, C112, etc. The environment variable ORACLE_SID is usually set to <DBSID>.
<SAPSID>	Specifies the name of the SAP system id. It is used for OS user names e.g. c11adm as well as for authentication of the OS user with Oracle e.g. OPS\$C11ADM.
<IHRDBMS>	RDBMS Software Installation Home. The <u>installation</u> Oracle Home directory where the Oracle RDBMS Software is installed. If you are running one database from the Oracle Home install the software to '/oracle/<DBNAME>/11202'. If you are running multiple databases from the same Oracle Home install the software to '/oracle/RDBMS/11202'.

<OHGRID>	<p>The path of the Oracle Home where the GRID Software was installed.</p> <p>Unix/Linux: '/oracle/GRID/11202', '/oracle/GRID/11203', ...</p> <p>Related environment variables: ORACLE_BASE='/oracle/BASE' (see SAP note #1521371) ORACLE_HOME='/oracle/GRID/11202' LD_LIBRARY_PATH or SHLIB_PATH='/oracle/GRID/11202/lib'</p> <p>Windows: 'C:\oracle\GRID\11202', 'C:\oracle\GRID\11203', ...</p>
<OHRDBMS>	<p>The <u>runtime</u> path of the Oracle Home where the RDBMS Software was installed. This is a symbolic link to <IHRDBMS>.</p> <p>Unix/Linux: If you are running one database from the Oracle Home: '/oracle/<DBNAME>/112' -> '/oracle/<DBNAME>/11202'</p> <p>If you are running multiple databases from the Oracle Home: '/oracle/<DBNAME>/112' -> '/oracle/RDBMS/11202'</p> <p>Related environment variables: ORACLE_BASE='/oracle' ORACLE_HOME='/oracle/<DBNAME>/112' LD_LIBRARY_PATH or SHLIB_PATH='/oracle/<DBNAME>/112/lib'</p> <p>Note: In cluster environments it is recommended to place <OHRDBMS> on a shared cluster file system.</p>
TABLESPACE SET	<p>A tablespace set is a number of 1..n tablespaces. The objects within a tablespace set (tables, indexes, etc.) do not have any references to other objects outside of the tablespace set. A tablespace set is “self contained”.</p>

TTS	“Transportable Tablespaces” is an Oracle technique that allows transporting a tablespace set from one database to another. During the transport it is possible to convert the datafiles from a “source” format to a “target” format. This may be required if source and target OS platforms are different, especially if the endianness between both is different.
-----	---

Oracle ASM Disk Group Configuration

For SAP on Oracle with Oracle ASM we recommend to choose one out of three variants on how to name and configure your Oracle ASM disk groups. Which variant you choose depends on your backup strategy which in turn usually depends on the size of your database and required availability.

Oracle ASM Redundancy

With Oracle ASM the database instance implements several levels of redundancy: external, normal and high.

External redundancy: The database instance never writes the same data to more than one failure group. If the underlying disks are not mirrored somehow a failure of a single disk means data loss. This mode is recommended for non-clustered environments where redundancy is achieved by the storage subsystem e.g. EMC or NetApp.

Normal redundancy: The database writes the same data to two different failure groups.

High redundancy: The database writes the same data to three different failure groups.

Please note that with Oracle ASM redundancy is achieved by the database instance by multiplexing an IO going to one disk (within a failure group) to one or two additional disks (within one or two other failure groups). This is different to traditional storage solutions where data redundancy is achieved by raid implementations where a piece of data needs to be written only once by the host and the IO bandwidth required for mirroring is provided by the storage system. Therefore it may be necessary to increase the IO bandwidth between server and storage or disks by using multipath software or by adding additional physical IO channels.

Example:

If you configure 2 redo log members per redo log group and place them on a Oracle ASM disk group with normal redundancy the database instance has to do 4 IO's - one for each redo log member and one for each disk.

Nevertheless it is not recommended to run your database with just one online redo log member per group!!!

Oracle ASM Disk Group Naming Conventions

Independent from how you configure your Oracle ASM disk groups for running your Oracle Database(s) it is recommended to configure one Oracle ASM disk group with “normal or high redundancy” exclusively for Oracle Clusterware. This Oracle ASM disk group stores the Oracle Cluster Repository and the Oracle ASM parameter file.

The recommended name for this Oracle ASM disk group is “+OCR”.

The minimal size of each Oracle ASM disk in the OCR disk group is 4GB. Note that each Oracle ASM disk defines a separate failure group within the disk group.

For RAC it is required to have a dedicated Oracle ASM disk group for the Oracle ASM Cluster File System ACFS. The disk group and the ACFS file system on it should be configured large enough to store the required number of Oracle RDBMS Homes. Keep in mind that much more disk space may be required than just for one Oracle Home when it comes to patching or out of place upgrades which are basically full installations since 11gR2 and need to be installed next to the old Oracle Home.

The recommended name for this Oracle ASM disk group is “+ACFS”.

Variant 1 – small to medium data volumes, restore time not critical

Customer has small or medium sized databases where backup and/or restore + recovery of **all** databases can be accomplished by RMAN in a timely fashion.

Example:

If a complete Oracle ASM disk group with data files from one or more databases is broken and all data files from all databases need to be restored to a newly created Oracle ASM disk group using RMAN.

Oracle ASM disk group recommendation:

Oracle ASM Disk Group Name	Stores
+DATA	<ul style="list-style-type: none"> - All data files - All temp files - Control file (first copy) - Online redo logs (first copy)
+ARCH	<ul style="list-style-type: none"> - Control file (second copy) - Archived redo logs
+RECO	<ul style="list-style-type: none"> - Control file (third copy) - Online redo logs (second copy) - RMAN backups (optional) - Fast recovery area (optional)

Variant 2 – medium to large data volumes, restore time important

Customer has medium to large sized databases where backup and/or restore + recovery of all databases cannot be accomplished in a timely fashion.

Usually customers will use RMAN and Split-Mirror techniques in combination.

Major differences to Variant 1 are:

- Separate Oracle ASM disk group for each database
- <DBNAME>+“_” are used as a prefix for the name of the DATA disk group.
- The number of the DATA disk group is appended if the database spans over more than one DATA disk group.
- No online redo logs are located in the “data” disk groups. Instead an additional disk group is used for the first member of each online redo log group.

Important: Please consider that only a maximum of 63 Oracle ASM disk groups can be configured.

Oracle ASM disk group recommendation:

Oracle ASM Disk Group Name	Stores
+<DBNAME>_DATA[#] e.g. ERP_DATA, ERP_DATA1, CRM_DATA1	<ul style="list-style-type: none"> - All data files - All temp files - Control file (first copy)
+OLOG	<ul style="list-style-type: none"> - Online redo logs (first copy)
+ARCH	<ul style="list-style-type: none"> - Control file (second copy) - Archived redo logs
+RECO	<ul style="list-style-type: none"> - Control file (third copy) - Online redo logs (second copy) - RMAN backups - Fast recovery area

Variant 3 – very large data and data change volumes, restore time crucial

Customer has very large databases where backup and/or restore + recovery of a single databases cannot be accomplished in a timely fashion.

Usually customers will use RMAN and Split-Mirror techniques in combination.

In this variant each relevant database file type is separated to different Oracle ASM disk groups.

Important: Please consider that only a maximum of 63 Oracle ASM disk groups can be configured.

Oracle ASM disk group recommendation:

Oracle ASM Disk Group Name	Stores
+<DBNAME>_DATA[#] e.g. ERP_DATA, ERP_DATA1, CRM_DATA	<ul style="list-style-type: none"> - All data files - All temp files - Control file (first copy)
+<DBNAME>_ARCH	<ul style="list-style-type: none"> - Control file (second copy)

e.g. ERP_ARCH, CRM_ARCH	- Archived redo logs
+<DBNAME>_RECO e.g. ERP_RECO, CRM_RECO	- Control file (third copy) - RMAN backups - Fast recovery area
+<DBNAME>_OLOG e.g. ERP_OLOG, CRM_OLOG	- Online redo logs (first copy)
+<DBNAME>_MLOG e.g. ERP_MLOG, CRM_MLOG	- Online redo logs (second copy)

Important: Always place all files from one database into the according Oracle ASM disk groups of that database. Never place a file from database “ERP” into a disk group that belongs to database “CRM”.

Oracle ASM Compatibility Attributes

The disk group attributes that determine compatibility are 'compatible.asm', 'compatible.rdbms' and 'compatible.advm'. The 'compatible.asm' and 'compatible.rdbms' attribute settings determine the minimum Oracle Database software version numbers that a system can use for Oracle ASM and the database instance types respectively. For example, if the Oracle ASM compatibility setting is 11.2, and RDBMS compatibility is set to 11.1, then the Oracle ASM software version must be at least 11.2, and the Oracle Database client software version must be at least 11.1. The 'compatible.advm' attribute determines where the Oracle ASM Dynamic Volume Manager feature can be used to create Oracle ASM volumes in disk groups.

Make sure that 'compatible.asm' and 'compatible.rdbms' are set to minimum '11.2.0.2.0' for all Oracle ASM disk groups.

Example:

```
[oracle@oracx3v3 ~]$ sqlplus
```

```
SQL*Plus: Release 11.2.0.3.0 Production on Fri Dec 10 15:20:12 2010
```

```
Copyright (c) 1982, 2010, Oracle. All rights reserved.
```

Enter user-name: / as sysoracle asm

Connected to:
 Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit
 Production
 With the Automatic Storage Management option

```
SQL> select dg.name as diskgroup, substr(a.name,1,18) as name,
substr(a.value,1,24) as value from v$asm_diskgroup dg,
v$asm_attribute a where dg.group_number = a.group_number and
a.name like '%compatible%';
```

DISKGROUP	NAME	VALUE
ACFS	compatible.asm	11.2.0.0.0
ACFS	compatible.rdbms	10.1.0.0.0
ARCH	compatible.asm	11.2.0.0.0
ARCH	compatible.rdbms	10.1.0.0.0
DATA	compatible.asm	11.2.0.0.0
DATA	compatible.rdbms	10.1.0.0.0
RECO	compatible.asm	11.2.0.0.0
RECO	compatible.rdbms	10.1.0.0.0
OCR	compatible.asm	11.2.0.0.0
OCR	compatible.rdbms	10.1.0.0.0

```
SQL> alter diskgroup ACFS set attribute 'compatible.asm' =
'11.2.0.2.0';
```

Diskgroup altered.

```
SQL> alter diskgroup ACFS set attribute 'compatible.rdbms' =
'11.2.0.2.0';
```

Diskgroup altered.

```
SQL> alter diskgroup ARCH set attribute 'compatible.asm' =
'11.2.0.2.0';
```

Diskgroup altered.

```
SQL> alter diskgroup ARCH set attribute 'compatible.rdbms' =
'11.2.0.2.0';
```

Diskgroup altered.

```
SQL> alter diskgroup DATA set attribute 'compatible.asm' =
'11.2.0.2.0';
```

Diskgroup altered.

```
SQL> alter diskgroup DATA set attribute 'compatible.rdbms' =
'11.2.0.2.0';
```

Diskgroup altered.

```
SQL> alter diskgroup RECO set attribute 'compatible.asm' =
'11.2.0.2.0';
```

Diskgroup altered.

```
SQL> alter diskgroup RECO set attribute 'compatible.rdbms' =
'11.2.0.2.0';
```

Diskgroup altered.

```
SQL> alter diskgroup OCR set attribute 'compatible.asm' =
'11.2.0.2.0';
```

Diskgroup altered.

```
SQL> alter diskgroup OCR set attribute 'compatible.rdbms' =
'11.2.0.2.0';
```

Diskgroup altered.

```
SQL> select dg.name as diskgroup, substr(a.name,1,18) as name,
substr(a.value,1,24) as value from v$asm_diskgroup dg,
v$asm_attribute a where dg.group_number = a.group_number and
a.name like '%compatible%';
```

DISKGROUP	NAME	VALUE
ACFS	compatible.asm	11.2.0.2.0
ACFS	compatible.rdbms	11.2.0.2.0
ARCH	compatible.asm	11.2.0.2.0
ARCH	compatible.rdbms	11.2.0.2.0
DATA	compatible.asm	11.2.0.2.0
DATA	compatible.rdbms	11.2.0.2.0
RECO	compatible.asm	11.2.0.2.0
RECO	compatible.rdbms	11.2.0.2.0
OCR	compatible.asm	11.2.0.2.0
OCR	compatible.rdbms	11.2.0.2.0

Prerequisites

Unix/Linux platforms

Primary group of the 'oracle' executable

After installing a new <OHRDBMS> the Oracle executable under <OHRDBMS>/bin is owned by OS user 'oracle' with 'oinstall' as its primary group.

```
-rwsr-s--x 1 oracle oinstall 228886468 Nov 29 15:29
/oracle/RM1/112/bin/oracle
```

To allow access to Oracle ASM the primary group needs to be changed by executing the script 'setasmgidwrap' under <OHGRID>/bin.

Example:

```
/oracle/GRID/11202/bin/setasmgidwrap o=/oracle/RM1/112/bin/oracle
```

```
[oracle@oracx3v3 dbs]$ ls -la /oracle/RM1/112/bin/oracle
-rwsr-s--x 1 oracle asmadmin 228886468 Nov 29 15:29
/oracle/RM1/112/bin/oracle
```

Migrating an Active Database from File System to Oracle ASM without Changing the OS platform

The process of creating a copy of an existing and potentially running database using RMAN is known as “duplicate from active database”. This type of operation is supported for source databases that are in “mount” or “open” state. This part of the document describes the prerequisites and the necessary steps to move an existing database to Oracle ASM using the “duplicate from active database” method.

This method can only be used if the following preconditions are met:

- If you are moving the database to a different host both hosts must be connected via a TCP/IP network.
- The source database must be on Oracle Database 11.2.0.2 or higher.
- The “compatible” parameter of the source database must be set to 11.2.0.2 or higher.
- If source and target do NOT run on the same platform both must run on a compatible platform. To consider platforms as compatible the following conditions must be met:

- Both platforms must use the same endian format. See view '\$TRANSPORTABLE_PLATFORM'.
- Only the following platform combinations are supported
 - Solaris-x64 <-> Linux-x64
 - HP-PA <-> HP-IA
 - Linux <-> Windows (requires minimum 11.1 compatible setting)
 - Solaris SPARC (64-bit) <-> AIX (64-bit) (source database must be non-RAC and non-TDE)

If the source database is in “open” state the target database requires recovery to be in a consistent state. This is done by RMAN automatically at the end of the duplicate operation.

Configuration of the Source and Target system

Setup the source system

On the source system only a few steps may be required to prepare the system for duplication.

Example environment for the ora<DBNAME> user:

```
ORACLE_SID=RM1
ORACLE_BASE=/oracle
PATH=./usr/kerberos/bin:/usr/local/bin:/bin:/usr/bin:
/oracle/RM1/112/bin
ORACLE_HOME=/oracle/RM1/112
```

MS Windows:

On MS Windows this is controlled by the path from where a program is executed. Ensure that your PATH always points to the correct Oracle Home before running an Oracle program like sqlplus, netca, etc.

Set at least ORACLE_SID and the PATH variable.

Example:

Set PATH to C:\ORACLE\GRID\112\bin and ORACLE_SID to +Oracle ASM when running GRID utilities like srvctl, crs_stat or sqlplus (connecting to the Oracle ASM instance).

Set PATH to C:\ORACLE\RM1\112\bin and ORACLE_SID=RM1 when running RDBMS utilities like oradim, orapwd or sqlplus (connecting to the DATABASE instance).

Oracle Password File

Ensure you have configured a password file for the instance.

Unix/Linux

The password file should be located under <OHRDBMS>/dbs. The default name of the password file is “orapw<DBSID>”

MS Windows

The password file should be located under <OHRDBMS>/database. The default name of the password file is “PWD<DBSID>.ora”

If you do not already have a password file configured use the orapwd utility to create a new one.

Example:

```
[oracle@oracx3v4 dbs]$ orapwd file=orapwRM1 entries=10
password=secret
[oracle@oracx3v4 dbs]$ ls -ltr
total 9728
-rw-r----- 1 oracle oinstall      24 Nov 30 13:20 lkRM1
-rw-r----- 1 oracle oinstall    2560 Nov 30 17:59 spfileRM1.ora
-rw-r----- 1 oracle oinstall    2560 Dec  1 15:22 orapwRM1
-rw-rw----  1 oracle oinstall    1544 Dec  8 13:40 hc_RM1.dat
```

SQL*Net configuration

File: "listener.ora"

```
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = RM1)
      (ORACLE_HOME = /oracle/RM1/112)
    )
  )

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP) (HOST =
oracx3v4.wdf.sap.corp) (PORT = 1527))
      (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1527))
    )
  )

ADR_BASE_LISTENER = /oracle
```

File: "tnsnames.ora"

```
ORGRM1 =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST =
oracx3v4.wdf.sap.corp) (PORT = 1527))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = RM1)
    )
  )

DUPRM1 =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST =
oracx3v3.wdf.sap.corp) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = RM1)
    )
  )
```

If your `sqlnet.ora` defines `NAMES.DEFAULT_DOMAIN` you will have to append the value to the tag in `tnsnames.ora` e.g. “DUPRM1.WORLD”.

File: “sqlnet.ora”

On MS Windows set:

```
SQLNET.AUTHENTICATION_SERVICES = (NTS)
```

Setup the target system

As already mentioned above it is assumed that you have properly installed and configured the following components:

- Clusterware and Oracle ASM (Oracle Grid Infrastructure)
- Oracle ASM disk groups for the database that will be copied to Oracle ASM
- New <OHRDBMS> for the database that will be copied.

Have a small shell script your “oracle” user can source or run for each Oracle Home that has been installed. Example: one for <OHGRID> and one for <OHRDBMS>.

Example environment script for user “oracle” set to work with the Oracle ASM instance or to run GRID commands:

```
ORACLE_BASE=/oracle/BASE
ORACLE_HOME=/oracle/GRID/11202
ORACLE_SID=+ASM
PATH=./usr/kerberos/bin:/usr/local/bin:/bin:/usr/bin:
/oracle/GRID/11202/bin
```

```
export ORACLE_BASE
export ORACLE_HOME
export ORACLE_SID
export PATH
```

Example environment script for user “oracle” set to work with the new database instance:

```
ORACLE_BASE=/oracle/RM1
ORACLE_HOME=/oracle/RM1/112
ORACLE_SID=RM1
PATH=./usr/kerberos/bin:/usr/local/bin:/bin:/usr/bin:
/oracle/RM1/112/bin
```

```
export ORACLE_BASE
export ORACLE_HOME
export ORACLE_SID
export PATH
```

In order to check your GRID setup run “crs_stat -t”.

Example:

```
[oracle@oracx3v3 ~]$ crs_stat -t
Name                Type                Target    State    Host
-----
ora.ACFS.dg         ora....up.type     ONLINE    ONLINE  oracx3v3
ora.ARCH.dg         ora....up.type     ONLINE    ONLINE  oracx3v3
ora.DATA.dg         ora....up.type     ONLINE    ONLINE  oracx3v3
ora.RECO.dg         ora....up.type     ONLINE    ONLINE  oracx3v3
ora....ER.lsnr     ora....er.type     ONLINE    ONLINE  oracx3v3
ora.OCR.dg          ora....up.type     ONLINE    ONLINE  oracx3v3
ora.asm             ora.asm.type       ONLINE    ONLINE  oracx3v3
ora.cssd            ora.cssd.type      ONLINE    ONLINE  oracx3v3
ora.diskmon         ora....on.type     ONLINE    ONLINE  oracx3v3
ora.evmd            ora.evm.type       ONLINE    ONLINE  oracx3v3
ora.ons             ora.ons.type       OFFLINE   OFFLINE
[oracle@oracx3v3 ~]$
```

Please note that 'crs_stat' is deprecated with Oracle Grid Infrastructure 11gR2. Instead we recommend to use 'crsctl status resource -t'.

SQL*Net

File “listener.ora”

```
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = RM1)
      (ORACLE_HOME = /oracle/RM1/112)
    )
  )

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))
      (ADDRESS = (PROTOCOL = TCP) (HOST =
oracx3v3.wdf.sap.corp) (PORT = 1521))
    )
  )

ADR_BASE_LISTENER = /oracle

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER=ON
```

File “tnsnames.ora”

```
ORGRM1 =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST =
oracx3v4.wdf.sap.corp) (PORT = 1527))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = RM1)
    )
  )

DUPRM1 =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST =
oracx3v3.wdf.sap.corp) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = RM1)
    )
  )
```

File “sqlnet.ora”

```
NAMES.DIRECTORY_PATH= (TNSNAMES, EZCONNECT)
ADR_BASE = /oracle
```

On MS Windows:

```
SQLNET.AUTHENTICATION_SERVICES = (NTS)
```

Password File

Create the same password file on the target system like you already have on the source system or create a new one with the same password for user “SYS”.

Additional configuration steps for MS Windows

Ensure that the user under which you install and configure your system is a domain user and member of the following local groups:

```
Administrators, ora_dba, ora_oper, ora_<DBNAME>_dba,
ora_<DBNAME>_oper
```

Create a new Windows service for your new instance

Example:

```
oradim -new -sid RM1
```

Checking your configuration

Checking network configuration from the source host

If you have followed the examples about SQL*Net configuration you should be able to connect from the source system to the remote system via sqlplus and to startup and shutdown the remote target database instance.

Example:

```
oracle@oracx3v4 dbs]$ sqlplus sys/secret@duprml as sysdba
```

```
SQL*Plus: Release 11.2.0.3.0 Production on Thu Dec 9 15:15:34
2010
```

```
Copyright (c) 1982, 2010, Oracle. All rights reserved.
```

```
Connected to an idle instance.
```

```
SQL> startup
```

```
ORA-32004: obsolete or deprecated parameter(s) specified for
RDBMS instance
ORACLE instance started.
```

```
Total System Global Area 1068937216 bytes
Fixed Size                  2233336 bytes
Variable Size               285215752 bytes
Database Buffers           775946240 bytes
Redo Buffers                 5541888 bytes
Database mounted.
Database opened.
SQL> shutdown
Database closed.
Database dismounted.
ORACLE instance shut down.
```

Cleanup Oracle ASM before duplicate

If there are any database files on Oracle ASM from earlier copy tests you should remove them before copying them again.

If you have a spfile (e.g. spfileRM1.ora) in your <OHRDBMS>/dbs or <OHRDBMS>\database directory you should delete or rename it before re-running the duplication process to ensure that the correct pfile is used for duplication.

Prepare target instance

Create a “pfile” on the target system

The next step before starting the duplication process is to startup the target instance into “nomount” state. Before this can be done you need to prepare an Oracle initialization parameter file (pfile) that contains all parameters required to startup the instance and to translate parameter values and file locations valid in the source system to parameter values valid in the target system. Place the file into directory <OHRDBMS>/dbs (or <OHRDBMS>\database on Windows).

Example: initRM1.ora:

```
*.control_file_record_keep_time=30
*.db_block_size=8192
*.db_domain='oracledev.wdf.sap.corp'
*.db_files=1500
*.db_name='RM1'
*.filesystemio_options='SETALL'
*.java_pool_size=0
*.log_archive_dest_1='location=+ARCH
valid_for=(ALL_LOGFILES,ALL_ROLES) db_unique_name=RM1'
db_unique_name='RM1'
*.log_archive_dest_state_1='enable'
```

```
*.log_archive_format='RM1ARC%t_%s_%r.dbf'
*.log_archive_max_processes=2
*.log_archive_min_succeed_dest=1
*.log_archive_trace=0
*.log_buffer=1048576
*.log_checkpoints_to_alert=TRUE
*.open_cursors=800
*.optimizer_index_caching=50
*.optimizer_index_cost_adj=20
*.os_authent_prefix='ops$'
*.PARALLEL_EXECUTION_MESSAGE_SIZE=16384
*.PARALLEL_MAX_SERVERS=80
*.parallel_min_servers=10
*.PARALLEL_THREADS_PER_CPU=1
*.pga_aggregate_target=1G
*.processes=850
*.QUERY_REWRITE_ENABLED='FALSE'
*.RECYCLEBIN='OFF'
*.remote_login_passwordfile='exclusive'
*.remote_os_authent=true
*.REPLICATION_DEPENDENCY_TRACKING=FALSE
*.sessions=1700
*.sga_target=1G
*.shared_pool_size=200M
*.star_transformation_enabled='true'
RM1.thread=1
*.timed_statistics=true
*.undo_management='AUTO'
RM1.undo_tablespace='PSAPUNDO'
*.db_create_file_dest='+DATA'
*.db_create_online_log_dest_1='+DATA'
*.db_create_online_log_dest_2='+RECO'
*.log_file_name_convert=
'/oracle/RM1/origlog/log_g1m1.dbf', '+DATA',
'/oracle/RM1/mirrlog/log_g1m2.dbf', '+RECO',
'/oracle/RM1/origlog/log_g2m1.dbf', '+DATA',
'/oracle/RM1/mirrlog/log_g2m2.dbf', '+RECO',
'/oracle/RM1/origlog/log_g3m1.dbf', '+DATA',
'/oracle/RM1/mirrlog/log_g3m2.dbf', '+RECO'
*.control_files=
'+DATA/RM1/cntrlRM1.dbf',
'+ARCH/RM1/cntrlRM1.dbf',
'+RECO/RM1/cntrlRM1.dbf'
*.compatible='11.2.0.2.0'
*.DB_RECOVERY_FILE_DEST='+RECO'
*.DB_RECOVERY_FILE_DEST_SIZE=4G
*._compression_compatibility='11.2.0'
```

Note: This pfile is only used for the duplication process and needs to be replaced by your own spfile that incorporates all the parameters settings recommended for your system.

Note: `*._compression_compatibility='11.2.0'` is required only if your source database uses the advanced compression option (ACO).

Start the target instance into “nomount” state

```
SQL> connect sys/secret@DUPRM1 as SYSDBA
Connected to an idle instance.
SQL> startup nomount
ORA-32004: obsolete or deprecated parameter(s) specified for
RDBMS instance
ORACLE instance started.

Total System Global Area 1068937216 bytes
Fixed Size                2182592 bytes
Variable Size             234881600 bytes
Database Buffers         830472192 bytes
Redo Buffers              1400832 bytes
SQL>
```

RMAN script for "duplicate from active database"

Create the following files in a working directory of your choice. They are required during the copy process.

Example: cprm1.rcv:

```
connect target sys/secret@ORGRM1
connect auxiliary sys/secret@DUPRM1
run {

ALLOCATE CHANNEL t1 DEVICE TYPE disk;
ALLOCATE CHANNEL t2 DEVICE TYPE disk;
ALLOCATE CHANNEL t3 DEVICE TYPE disk;
ALLOCATE CHANNEL t4 DEVICE TYPE disk;

ALLOCATE AUXILIARY CHANNEL a1 DEVICE TYPE disk;

duplicate target database
          to RM1
          from active database
nofilenamecheck;
}
```

The number of channels (allocated by the 'allocate channel' command) that should be allocated depends on the available bandwidth of your network connection. For a standard 1gigabit Ethernet interface we recommend at least 4 channels.

The number of auxiliary channels is currently limited to 1.

Copying the database

Start the source database into "mount" or "open" state.

When you copy from an open database you need enough free disk space to keep all the archived redo logs written by the source database instance during the duplication process. Remember that the amount of disk space required for archived redo logs depends on the workload and the time needed to copy the database to the destination.

Run "RMAN duplicate from active database"

Example:

```
Recovery Manager: Release 11.2.0.3.0 - Production on Mon Dec 6
16:35:51 2010
```

```
Copyright (c) 1982, 2009, Oracle and/or its affiliates. All
rights reserved.
```

```
RMAN>
```

```
RMAN> connect target *
connected to target database: RM1 (DBID=986144351)
```

```
RMAN> connect auxiliary *
connected to auxiliary database: RM1 (not mounted)
```

```
RMAN> run {
2>
3> ALLOCATE CHANNEL t1 DEVICE TYPE disk;
4> ALLOCATE CHANNEL t2 DEVICE TYPE disk;
5> ALLOCATE CHANNEL t3 DEVICE TYPE disk;
6> ALLOCATE CHANNEL t4 DEVICE TYPE disk;
7>
8> ALLOCATE AUXILIARY CHANNEL a1 DEVICE TYPE disk;
9>
10> duplicate target database
11>          to RM1
12>          from active database
13> nofilenamecheck;
14>}
```

```
using target database control file instead of recovery catalog
allocated channel: t1
channel t1: SID=8 device type=DISK
```

```
allocated channel: t2
channel t2: SID=579 device type=DISK
```

```
allocated channel: t3
channel t3: SID=1145 device type=DISK
```

```
allocated channel: t4
channel t4: SID=1716 device type=DISK
```

```
allocated channel: a1
channel a1: SID=426 device type=DISK
Starting Duplicate Db at 06-DEC-10
```

```
contents of Memory Script:
{
  sql clone "create spfile from memory";
}
executing Memory Script
```

```
sql statement: create spfile from memory
```

```
contents of Memory Script:
{
  shutdown clone immediate;
  startup clone nomount;
}
executing Memory Script
```

```
Oracle instance shut down
```

```
connected to auxiliary database (not started)
Oracle instance started
```

```
Total System Global Area      1068937216 bytes
```

```
Fixed Size                      2233336 bytes
Variable Size                    260049928 bytes
Database Buffers                  801112064 bytes
Redo Buffers                       5541888 bytes
```

```
allocated channel: a1
channel a1: SID=431 device type=DISK
```

```
contents of Memory Script:
{
  sql clone "alter system set db_name =
  'RM1' comment="
```

```
'Modified by RMAN duplicate'' scope=spfile";
  sql clone "alter system set db_unique_name =
'MR1'' comment=
'Modified by RMAN duplicate'' scope=spfile";
  shutdown clone immediate;
  startup clone force nomount
  backup as copy current controlfile auxiliary format
'+DATA/RM1/cntrlRM1.dbf';
  restore clone controlfile to '+ARCH/RM1/cntrlRM1.dbf' from
'+DATA/RM1/cntrlRM1.dbf';
  restore clone controlfile to '+RECO/RM1/cntrlRM1.dbf' from
'+DATA/RM1/cntrlRM1.dbf';
  alter clone database mount;
}
```

executing Memory Script

```
sql statement: alter system set db_name = 'MR1' comment=
'Modified by RMAN duplicate'' scope=spfile
```

```
sql statement: alter system set db_unique_name = 'MR1'
comment= 'Modified by RMAN duplicate'' scope=spfile
```

Oracle instance shut down

Oracle instance started

Total System Global Area 1068937216 bytes

Fixed Size 2233336 bytes

Variable Size 260049928 bytes

Database Buffers 80112064 bytes

Redo Buffers 5541888 bytes

allocated channel: a1

channel a1: SID=426 device type=DISK

Starting backup at 06-DEC-10

channel t1: starting datafile copy

copying current control file

output file name=/oracle/RM1/112/dbs/snapcf_RM1.f

tag=TAG20101206T163622 RECID=9 STAMP=737051782

channel t1: datafile copy complete, elapsed time: 00:00:07

Finished backup at 06-DEC-10

Starting restore at 06-DEC-10

channel a1: copied control file copy

Finished restore at 06-DEC-10

Starting restore at 06-DEC-10

```
channel a1: copied control file copy
Finished restore at 06-DEC-10
database mounted
```

```
contents of Memory Script:
```

```
{
  set newname for clone datafile 1 to new;
  set newname for clone datafile 2 to new;
  set newname for clone datafile 3 to new;
  set newname for clone datafile 4 to new;
  backup as copy reuse
  datafile 1 auxiliary format new
  datafile 2 auxiliary format new
  datafile 3 auxiliary format new
  datafile 4 auxiliary format new
  ;
  sql 'alter system archive log current';
}
```

```
executing Memory Script
```

```
executing command: SET NEWNAME
```

```
Starting backup at 06-DEC-10
channel t1: starting datafile copy
input datafile file number=00001
name=/oracle/RM1/sapdata1/system_1/system.data1
channel t2: starting datafile copy
input datafile file number=00002
name=/oracle/RM1/sapdata1/sysaux_1/sysaux.data1
channel t3: starting datafile copy
input datafile file number=00003
name=/oracle/RM1/sapdata2/undo_1/undo.data1
channel t4: starting datafile copy
input datafile file number=00004
name=/oracle/RM1/sapdata2/stabd_1/stabd.data1
output file name=+DATA/rm1/datafile/psapstabd.260.737052085
tag=TAG20101206T163636
channel t4: datafile copy complete, elapsed time: 00:02:47
output file name=+DATA/rm1/datafile/psapundo.261.737052085
tag=TAG20101206T163636
channel t3: datafile copy complete, elapsed time: 00:04:08
output file name=+DATA/rm1/datafile/sysaux.262.737052083
tag=TAG20101206T163636
channel t2: datafile copy complete, elapsed time: 00:11:08
```

```
output file name=+DATA/rm1/datafile/system.263.737052083
tag=TAG20101206T163636
channel t1: datafile copy complete, elapsed time: 00:11:38
Finished backup at 06-DEC-10
```

```
sql statement: alter system archive log current
```

```
contents of Memory Script:
```

```
{
  backup as copy reuse
  archivelog like
"/oracle/RM1/oraarch/RM1ARC1_36_736435231.dbf" auxiliary format
"/oracle/RM1/oraarch/RM1ARC1_36_736435231.dbf" archivelog like
"/oracle/RM1/oraarch/RM1ARC1_37_736435231.dbf" auxiliary format
"/oracle/RM1/oraarch/RM1ARC1_37_736435231.dbf" ;
  catalog clone archivelog
"/oracle/RM1/oraarch/RM1ARC1_36_736435231.dbf";
  catalog clone archivelog
"/oracle/RM1/oraarch/RM1ARC1_37_736435231.dbf";
  switch clone datafile all;
}
```

```
executing Memory Script
```

```
Starting backup at 06-DEC-10
channel t1: starting archived log copy
input archived log thread=1 sequence=36 RECID=24 STAMP=737051822
channel t2: starting archived log copy
input archived log thread=1 sequence=37 RECID=25 STAMP=737052495
output file name=/oracle/RM1/oraarch/RM1ARC1_36_736435231.dbf
RECID=0 STAMP=0
channel t1: archived log copy complete, elapsed time: 00:00:01
output file name=/oracle/RM1/oraarch/RM1ARC1_37_736435231.dbf
RECID=0 STAMP=0
channel t2: archived log copy complete, elapsed time: 00:00:01
Finished backup at 06-DEC-10
```

```
cataloged archived log
archived log file
name=/oracle/RM1/oraarch/RM1ARC1_36_736435231.dbf RECID=24
STAMP=737052791
```

```
cataloged archived log
archived log file
name=/oracle/RM1/oraarch/RM1ARC1_37_736435231.dbf RECID=25
STAMP=737052791
```

```
datafile 1 switched to datafile copy
input datafile copy RECID=9 STAMP=737052792 file
name=+DATA/rm1/datafile/system.263.737052083
datafile 2 switched to datafile copy
```

```
input datafile copy RECID=10 STAMP=737052792 file
name=+DATA/rml/datafile/sysaux.262.737052083
datafile 3 switched to datafile copy
input datafile copy RECID=11 STAMP=737052792 file
name=+DATA/rml/datafile/psapundo.261.737052085
datafile 4 switched to datafile copy
input datafile copy RECID=12 STAMP=737052792 file
name=+DATA/rml/datafile/psapstabd.260.737052085
```

contents of Memory Script:

```
{
  set until scn 284217;
  recover
  clone database
  delete archivelog
  ;
}
```

executing Memory Script

executing command: SET until clause

Starting recover at 06-DEC-10

starting media recovery

```
archived log for thread 1 with sequence 36 is already on disk as
file /oracle/RM1/oraarch/RM1ARC1_36_736435231.dbf
archived log for thread 1 with sequence 37 is already on disk as
file /oracle/RM1/oraarch/RM1ARC1_37_736435231.dbf
archived log file
name=/oracle/RM1/oraarch/RM1ARC1_36_736435231.dbf thread=1
sequence=36
archived log file
name=/oracle/RM1/oraarch/RM1ARC1_37_736435231.dbf thread=1
sequence=37
media recovery complete, elapsed time: 00:00:01
Finished recover at 06-DEC-10
Oracle instance started
```

Total System Global Area 1068937216 bytes

Fixed Size	2233336 bytes
Variable Size	276827144 bytes
Database Buffers	784334848 bytes
Redo Buffers	5541888 bytes

contents of Memory Script:

```
{
  sql clone "alter system set db_name =
  'RM1' comment="
```

```
'Reset to original value by RMAN'' scope=spfile";
  sql clone "alter system reset db_unique_name scope=spfile";
  shutdown clone immediate;
  startup clone nomount;
}
```

executing Memory Script

```
sql statement: alter system set db_name = 'RM1' comment=
'Reset to original value by RMAN'' scope=spfile
```

```
sql statement: alter system reset db_unique_name scope=spfile
```

Oracle instance shut down

```
connected to auxiliary database (not started)
Oracle instance started
```

```
Total System Global Area      1068937216 bytes
```

```
Fixed Size                      2233336 bytes
```

```
Variable Size                   276827144 bytes
```

```
Database Buffers               784334848 bytes
```

```
Redo Buffers                    5541888 bytes
```

```
allocated channel: a1
```

```
channel a1: SID=426 device type=DISK
```

```
sql statement: CREATE CONTROLFILE REUSE SET DATABASE "RM1"
```

```
RESETLOGS ARCHIVELOG
```

```
  MAXLOGFILES      16
```

```
  MAXLOGMEMBERS    3
```

```
  MAXDATAFILES     100
```

```
  MAXINSTANCES     8
```

```
  MAXLOGHISTORY    292
```

```
LOGFILE
```

```
  GROUP 1 ( '+DATA', '+RECO' ) SIZE 50 M REUSE,
```

```
  GROUP 2 ( '+DATA', '+RECO' ) SIZE 50 M REUSE,
```

```
  GROUP 3 ( '+DATA', '+RECO' ) SIZE 50 M REUSE
```

```
DATAFILE
```

```
  '+DATA/rm1/datafile/system.263.737052083'
```

```
CHARACTER SET UTF8
```

contents of Memory Script:

```
{
  set newname for clone tempfile 1 to new;
  switch clone tempfile all;
  catalog clone datafilecopy
"+DATA/rm1/datafile/sysaux.262.737052083",
"+DATA/rm1/datafile/psapundo.261.737052085",
"+DATA/rm1/datafile/psapstabd.260.737052085";
  switch clone datafile all;
}
```

```
executing Memory Script

executing command: SET NEWNAME

renamed tempfile 1 to +DATA in control file

cataloged datafile copy
datafile copy file name=+DATA/rm1/datafile/sysaux.262.737052083
RECID=1 STAMP=737052813
cataloged datafile copy
datafile copy file name=+DATA/rm1/datafile/psapundo.261.737052085
RECID=2 STAMP=737052813
cataloged datafile copy
datafile copy file
name=+DATA/rm1/datafile/psapstabd.260.737052085 RECID=3
STAMP=737052813

datafile 2 switched to datafile copy
input datafile copy RECID=1 STAMP=737052813 file
name=+DATA/rm1/datafile/sysaux.262.737052083
datafile 3 switched to datafile copy
input datafile copy RECID=2 STAMP=737052813 file
name=+DATA/rm1/datafile/psapundo.261.737052085
datafile 4 switched to datafile copy
input datafile copy RECID=3 STAMP=737052813 file
name=+DATA/rm1/datafile/psapstabd.260.737052085
contents of Memory Script:
{
  Alter clone database open resetlogs;
}
executing Memory Script
database opened
Finished Duplicate Db at 06-DEC-10
released channel: t1
released channel: t2
released channel: t3
released channel: t4
released channel: a1
RMAN>
RMAN> **end-of-file**
```

Post Duplication Tasks

Checks

Check the filenames and locations

After the duplication is finished successfully the target database instance is still up and running. Check if the filenames all include the correct Oracle ASM disk group in their name to be sure every file was copied to the right location.

Example:

```
[oracle@oracx3v3 dbs]$ sqlplus
```

```
SQL*Plus: Release 11.2.0.3.0 Production on Wed Dec 8 14:01:28
2010
```

```
Copyright (c) 1982, 2010, Oracle. All rights reserved.
```

```
Enter user-name: / as sysdba
```

```
Connected to:
```

```
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit
Production
```

```
With the Partitioning, Automatic Storage Management, OLAP, Data
Mining and Real Application Testing options
```

```
SQL> select name from v$datafile;
```

```
NAME
```

```
-----
+DATA/rm1/datafile/system.262.737214345
+DATA/rm1/datafile/sysaux.263.737214345
+DATA/rm1/datafile/psapundo.264.737214345
+DATA/rm1/datafile/psapstabd.261.737214345
```

```
SQL> select name from v$controlfile;
```

```
NAME
```

```
-----
+DATA/RM1/cntrlRM1.dbf
+ARCH/RM1/cntrlRM1.dbf
+RECO/RM1/cntrlRM1.dbf
```

```
SQL> select member from v$logfile;
```

```
MEMBER
```

```
-----
+DATA/RM1/onlinelog/group_3.257.737215021
+RECO/RM1/onlinelog/group_3.260.737215021
```

```
+DATA/RM1/onlinelog/group_2.258.737215019
+RECO/RM1/onlinelog/group_2.259.737215019
+DATA/RM1/onlinelog/group_1.259.737215017
+RECO/RM1/onlinelog/group_1.258.737215019
```

6 rows selected.

SQL>

Database Validation with RMAN

Ensure that all files are consistent and that no blocks are corrupted.

Example:

```
RMAN> connect target /
```

```
connected to target database: RM1 (DBID=986924135)
```

```
RMAN> validate database;
```

```
Starting validate at 08-DEC-10
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=432 device type=DISK
channel ORA_DISK_1: starting validation of datafile
channel ORA_DISK_1: specifying datafile(s) for validation
input datafile file number=00001
name=+DATA/rm1/datafile/system.262.737214345
input datafile file number=00002
name=+DATA/rm1/datafile/sysaux.263.737214345
input datafile file number=00003
name=+DATA/rm1/datafile/psapundo.264.737214345
input datafile file number=00004
name=+DATA/rm1/datafile/psapstabd.261.737214345
channel ORA_DISK_1: validation complete, elapsed time: 00:00:55
List of Datafiles
```

```
=====
File Status Marked Corrupt Empty Blocks Blocks Examined High SCN
-----
1      OK      0              79853          102400          333378
File Name: +DATA/rm1/datafile/system.262.737214345
Block Type Blocks Failing Blocks Processed
-----
Data      0              14941
Index     0              4644
Other     0              2962
```

```
File Status Marked Corrupt Empty Blocks Blocks Examined High SCN
```

```

-----
2    OK    0          81154          89600          333366
  File Name: +DATA/rm1/datafile/sysaux.263.737214345
  Block Type Blocks Failing Blocks Processed
-----
  Data      0          2325
  Index     0          1987
  Other     0          4134

```

```

File Status Marked Corrupt Empty Blocks Blocks Examined High SCN
-----
3    OK    0          1537          25600          333378
  File Name: +DATA/rm1/datafile/psapundo.264.737214345
  Block Type Blocks Failing Blocks Processed
-----
  Data      0          0
  Index     0          0
  Other     0          24063

```

```

File Status Marked Corrupt Empty Blocks Blocks Examined High SCN
-----
4    OK    0          12673          12800          18504
  File Name: +DATA/rm1/datafile/psapstabd.261.737214345
  Block Type Blocks Failing Blocks Processed
-----
  Data      0          0
  Index     0          0
  Other     0          127

```

```

channel ORA_DISK_1: starting validation of datafile
channel ORA_DISK_1: specifying datafile(s) for validation
including current control file for validation
including current SPFILE in backup set
channel ORA_DISK_1: validation complete, elapsed time: 00:00:01
List of Control File and SPFILE
=====

```

```

File Type      Status Blocks Failing Blocks Examined
-----
SPFILE         OK      0          2
Control File   OK      0          614
Finished validate at 08-DEC-10

```

RMAN>

Processing time of the 'validate database' command can be improved by allocating multiple channels. For maximum parallelism we recommend to allocate NUMBER_OF_CPU_CORES / 2 channels before executing the 'validate database' command. Example:

```

ALLOCATE CHANNEL t1 DEVICE TYPE disk;
ALLOCATE CHANNEL t2 DEVICE TYPE disk;
.
.
ALLOCATE CHANNEL t[n] DEVICE TYPE disk;

```

Server Parameter File

The next step is to create the final server parameter file (spfile).

Create a new pfile without all the initialization parameters that were required during the duplication process to translate names or locations and try to startup your database with it. Example:

```

initRM1_newprod.ora:
*.control_file_record_keep_time=30
*.db_block_size=8192
*.db_domain='oracledev.wdf.sap.corp'
*.db_files=1500
*.db_name='RM1'
*.filesystemio_options='SETALL'
*.java_pool_size=0
*.log_archive_dest_1='location=+ARCH
valid_for=(ALL_LOGFILES,ALL_ROLES)'
*.db_unique_name='RM1'
*.log_archive_dest_state_1='enable'
*.log_archive_format='RM1ARC%t_%s_%r.dbf'
*.log_archive_max_processes=2
*.log_archive_min_succeed_dest=1
*.log_archive_trace=0
*.log_buffer=1048576
*.log_checkpoints_to_alert=TRUE
*.open_cursors=800
*.optimizer_index_caching=50
*.optimizer_index_cost_adj=20
*.os_authent_prefix='ops$'
*.PARALLEL_EXECUTION_MESSAGE_SIZE=16384
*.PARALLEL_MAX_SERVERS=80
*.parallel_min_servers=10
*.PARALLEL_THREADS_PER_CPU=1
*.pga_aggregate_target=1G
*.processes=850
*.QUERY_REWRITE_ENABLED='FALSE'
*.RECYCLEBIN='OFF'
*.remote_login_passwordfile='exclusive'
*.remote_os_authent=true
*.REPLICATION_DEPENDENCY_TRACKING=FALSE
*.sessions=1700
*.sga_target=1G

```

```

*.shared_pool_size=200M
*.star_transformation_enabled='true'
RM1.thread=1
*.timed_statistics=true
*.undo_management='AUTO'
RM1.undo_tablespace='PSAPUNDO'
*.db_create_file_dest='+DATA'
*.db_create_online_log_dest_1='+DATA'
*.db_create_online_log_dest_2='+RECO'
*.control_files='+DATA/RM1/cntrlRM1.dbf','+ARCH/RM1/cntrlRM1.dbf'
,'+RECO/RM1/cntrlRM1.dbf'
*.compatible='11.2.0.2.0'
*.DB_RECOVERY_FILE_DEST='+RECO'
*.DB_RECOVERY_FILE_DEST_SIZE=4G
*._compression_compatibility='11.2.0'

```

Create the spfile

For all system types that run with ASM the standard location for the spfile is on ASM and not on a local filesystem or a cluster filesystem.

Example:

```
create spfile='+DATA/RM1/spfileRM1.ora' from
pfile='initRM1_newprod.ora';
```

Note that 'spfileRM1.ora' is just an alias on Oracle ASM:

```
Oracle ASMCMD> ls -l
Type          Redund  Striped   Sys  Name
              N       spfileRM1.ora =>
+DATA/RM1/PARAMETERFILE/spfile.267.737217857
```

Make sure that the database instance is least in "nomount" state (started with a pfile) before you create the final spfile on ASM. This ensures that the file is created at the correct location. Otherwise the file would be created under '+<DISKGROUP>/DB_UNKNOWN/PARAMETERFILE/...'.

There may be reasons where customers want to place the spfile on a local filesystem or on a cluster filesystem. This is supported and the spfile could be created under <OHRDBMS>/dbs (or <OHRDBMS>\database on MS Windows).

Example:

```
SQL> create spfile='spfileRM1.ora' from
pfile='initRM1_newprod.ora';
```

Although it is recommended that you use 'srvctl' to startup and shutdown your database it may be necessary to startup and shutdown the database with 'sqlplus' also. In this case you will need an additional pfile under <OHRDBMS>/dbs (or <OHRDBMS>\database on MS Windows) that points to the spfile on Oracle ASM.

Example:

```
initRM1.ora:
SPFILE='+DATA/RM1/SPFILERM1.ORA'
```

Register the database to the GRID

Registering the newly copied database to the GRID is the last step and is required to use 'srvctl' to startup and shutdown your database as well as to allow GRID to automatically startup your database (e.g. after reboot) together with all the resources it depends on (e.g. Oracle ASM disk groups) and all the resources that depend on the database (e.g. database services).

```
srvctl add database -d RM1 -o /oracle/RM1/112 -p
/oracle/RM1/112/dbs/spfilerM1.ora -a "ACFS,DATA,RECO,ARCH"
```

```
[oracle@oracx3v3 ~]$ crs_stat -t
```

Name	Type	Target	State	Host
ora.ACFS.dg	ora....up.type	ONLINE	ONLINE	oracx3v3
ora.ARCH.dg	ora....up.type	ONLINE	ONLINE	oracx3v3
ora.DATA.dg	ora....up.type	ONLINE	ONLINE	oracx3v3
ora.RECO.dg	ora....up.type	ONLINE	ONLINE	oracx3v3
ora....ER.lsnr	ora....er.type	ONLINE	ONLINE	oracx3v3
ora.OCR.dg	ora....up.type	ONLINE	ONLINE	oracx3v3
ora.asm	ora.asm.type	ONLINE	ONLINE	oracx3v3
ora.cssd	ora.cssd.type	ONLINE	ONLINE	oracx3v3
ora.diskmon	ora....on.type	ONLINE	ONLINE	oracx3v3
ora.evmd	ora.evm.type	ONLINE	ONLINE	oracx3v3
ora.ons	ora.ons.type	OFFLINE	OFFLINE	
ora.rm1.db	ora....se.type	OFFLINE	OFFLINE	

```
[oracle@oracx3v3 ~]$
```

Start the database

```
[oracle@oracx3v3 ~]$ srvctl start database -d RM1
```

```
[oracle@oracx3v3 ~]$ crs_stat -t
```

Name	Type	Target	State	Host
ora.ACFS.dg	ora....up.type	ONLINE	ONLINE	oracx3v3
ora.ARCH.dg	ora....up.type	ONLINE	ONLINE	oracx3v3
ora.DATA.dg	ora....up.type	ONLINE	ONLINE	oracx3v3
ora.RECO.dg	ora....up.type	ONLINE	ONLINE	oracx3v3
ora....ER.lsnr	ora....er.type	ONLINE	ONLINE	oracx3v3
ora.OCR.dg	ora....up.type	ONLINE	ONLINE	oracx3v3

ora.asm	ora.asm.type	ONLINE	ONLINE	oracx3v3
ora.cssd	ora.cssd.type	ONLINE	ONLINE	oracx3v3
ora.diskmon	ora....on.type	ONLINE	ONLINE	oracx3v3
ora.evmd	ora.evm.type	ONLINE	ONLINE	oracx3v3
ora.ons	ora.ons.type	OFFLINE	OFFLINE	
ora.rml.db	ora....se.type	ONLINE	ONLINE	oracx3v3

As already mentioned 'crs_stat' is still supported but deprecated with Oracle Grid Infrastructure 11gR2. Alternatively you can use 'crsctl status resource -t'.

Note that the example above is for registering a single instance database to the grid infrastructure. For Oracle RAC you may have to add more parameters like the names of the nodes or the management policy to the 'srvctl add database...' command.

Offline Database Migration from File System to Oracle ASM using Transportable Tablespaces

The following part of this document demonstrates the migration of a SAP database on file system from AIX 5.3 (Big Endian) to Oracle ASM on Oracle Enterprise Linux 5.5 x86_64 (Little Endian). Beside export/import strategies and online migration solutions like Oracle GoldenGate or Triple-O you can use Oracle Transportable Tablespaces (TTS) for homogeneous and heterogeneous OS platform migrations. Performing the migration using TTS is much easier and faster than using export/import. One disadvantage of export/import strategies as well as TTS is that both techniques require system downtime.

In the following examples both systems are connected to the same network via TCP/IP and the target system has NFS mounted the file systems with all the database files from the source system.

The migration is performed by transporting all SAP owned objects using the TTS technique. This technique allows you to define a set of tablespaces called the "tablespace set". A tablespace set can include all tablespaces **except of system-, undo- and temporary tablespaces** and can be migrated from any supported source platform to any supported target platform.

One of the most important requirements when using TTS is that the tablespace set needs to be self contained. This means that no object within the tablespace set may have references to objects that are not included in the same tablespace set. For more detailed information on restrictions or features of TTS refer to the Oracle software documentation for Oracle Database 11gR2.

Before starting with TTS make sure you have installed at least SAP RDBMS Bundle Patch 2012/08 for 11.2.0.2 or 11.2.0.3 as it contains a fix for Oracle bug 13001379.

The major steps of OS platform migrations with TTS are:

- Identify all the tablespaces and datafiles that need to be transported.
- Check whether your tablespace set is self-contained.
- Create RMAN scripts for datafile conversion on target platform
- Export of metadata of all database objects owned by the “SAP” user and the “OPSS\$<SAPSID>ADM” user (e.g. SAPSR3, OPSS\$RM1ADM).
- Ensure that the file system(s) with the original database are accessible on the target system by using NFS mounts.
- Use RMAN’s data file conversion feature to copy (file system to Oracle ASM) and convert (big endian to little endian) all datafiles from all tablespaces.
- Create a new database with SYSTEM, SYSAUX, UNDO and TEMP tablespaces on Oracle ASM.
- Plug-in the transported tablespaces into the newly created database.

Steps on the source platform

Supported platforms for TTS

Before you start your migration project make sure that your source and target platforms are supported for TTS by checking if both platforms are listed in view 'v\$transportable_platform'. If your platforms are not listed there you cannot use TTS for your migration project.

```
SQL> set linesize 10000
SQL> set pagesize 10000
SQL> select * from v$transportable_platform;
```

PLATFORM_ID	PLATFORM_NAME	ENDIAN_FORMAT
1	Solaris[tm] OE (32-bit)	Big
2	Solaris[tm] OE (64-bit)	Big
7	Microsoft Windows IA (32-bit)	Little
10	Linux IA (32-bit)	Little
6	AIX-Based Systems (64-bit)	Big
3	HP-UX (64-bit)	Big
5	HP Tru64 UNIX	Little
4	HP-UX IA (64-bit)	Big
11	Linux IA (64-bit)	Little
15	HP Open VMS	Little
8	Microsoft Windows IA (64-bit)	Little

9 IBM zSeries Based Linux	Big
13 Linux x86 64-bit	Little
16 Apple Mac OS	Big
12 Microsoft Windows x86 64-bit	Little
17 Solaris Operating System (x86)	Little
18 IBM Power Based Linux	Big
19 HP IA Open VMS	Little
20 Solaris Operating System (x86-64)	Little
21 Apple Mac OS (x86-64)	Little

Identify all tablespaces and datafiles that need to be migrated

```
SQL> set linesize 1000;
SQL> set pagesize 1000;
SQL> select tablespace_name,file_name from dba_data_files order
by tablespace_name,file_name;
```

TABLESPACE_NAME	FILE_NAME
PSAPSR3	/oracle/RM1/sapdata2/sr3_1/sr3.data1
PSAPSR3700	/oracle/RM1/sapdata2/sr3700_1/sr3700.data1
PSAPSR3USR	/oracle/RM1/sapdata2/sr3usr_1/sr3usr.data1
PSAPUNDO	/oracle/RM1/sapdata2/undo_1/undo.data1
SYSAUX	/oracle/RM1/sapdata1/sysaux_1/sysaux.data1
SYSTEM	/oracle/RM1/sapdata1/system_1/system.data1

6 rows selected

Change default tablespace of user SAPSR3 to 'SYSTEM'

```
SQL> alter user sapsr3 default tablespace system;
```

User altered.

This is required to be able to import the user into the target database while it's default tablespace was not yet created. It will be changed back later.

Move table 'SAPUSER' to a transportable tablespace

The table 'SAPUSER' owned by Oracle user 'OPS\$<SAPSID>ADM' is usually located in the SYSTEM tablespace which is not a transportable tablespace. Make sure you move this table to a tablespace that can be transported.

Example:

```
SQL> select tablespace_name from dba_tables where
table_name='SAPUSER';
```

```
TABLESPACE_NAME
```

```
-----  
SYSTEM
```

```
SQL> alter table ops$rmladm.sapuser move tablespace psapsr3usr;
```

```
Table altered.
```

```
SQL> select tablespace_name from dba_tables where  
table_name='SAPUSER';
```

```
TABLESPACE_NAME
```

```
-----  
PSAPSR3USR
```

Check if tablespace set is self contained

The following PL/SQL script is just an example script that collects the names from all tablespaces except tablespaces that cannot be transported with TTS.

Example: ttscheck.sql

```
set serveroutput on;
```

```
declare  
  tslist varchar2(32767);  
  first boolean:=true;  
begin  
  for c in (select tablespace_name from dba_tablespaces where  
            contents='PERMANENT' and tablespace_name != 'SYSTEM'  
and tablespace_name != 'SYSAUX') loop  
    if first = true then  
      first:=false;  
    else  
      tslist:=tslist||',';'  
    end if;  
    tslist:=tslist||c.tablespace_name;  
  end loop;  
  
  dbms_output.put_line('Checking: '||tslist);  
  dbms_tts.transport_set_check(to_clob(tslist), false, false);  
  
  for c in (select violations from transport_set_violations) loop  
    dbms_output.put_line(c.violations);  
  end loop;  
  
end;  
/
```

Run the script:

```
SQL> @ttscheck
Checking: PSAPSR3,PSAPSR3700,PSAPSR3USR

PL/SQL procedure successfully completed.
```

If no errors or warnings are displayed the tablespace set is self contained and can be transported.

Create RMAN script for conversion of datafiles on the target platform

Startup database in read only mode

```
SQL> startup mount
ORACLE instance started.

Total System Global Area 1219334144 bytes
Fixed Size                  2219088 bytes
Variable Size               1073742768 bytes
Database Buffers            134217728 bytes
Redo Buffers                 9154560 bytes
Database mounted.
```

```
SQL> alter database open read only;

Database altered.
```

Create conversion script with RMAN

```
$ cat cre_conv_script.rman
convert database on target platform
  convert script '/oracle/conv.rman'
  new database 'RM1'
  format '+DATA';
```

```
$ rman target /
```

```
Recovery Manager: Release 11.2.0.3.0 - Production on Tue Dec 21
13:25:27 2010
```

```
Copyright (c) 1982, 2009, Oracle and/or its affiliates. All
rights reserved.
```

```
connected to target database: RM1 (DBID=987524547)
```

```
RMAN> @cre_conv_script.rman
```

```
RMAN> convert database on target platform
2>  convert script '/oracle/conv.rman'
3>  new database 'RM1'
4>  format '+DATA';
Starting conversion at source at 21-DEC-10
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=1712 device type=DISK

Directory SYS.ORACLE_OCM_CONFIG_DIR found in the database
Directory SYS.DATA_PUMP_DIR found in the database

User SYS with SYSDBA and SYSOPER privilege found in password file
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00001
name=/oracle/RM1/sapdata1/system_1/system.data1
channel ORA_DISK_1: datafile checking complete, elapsed time:
00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00002
name=/oracle/RM1/sapdata1/sysaux_1/sysaux.data1
channel ORA_DISK_1: datafile checking complete, elapsed time:
00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00003
name=/oracle/RM1/sapdata2/undo_1/undo.data1
channel ORA_DISK_1: datafile checking complete, elapsed time:
00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00004
name=/oracle/RM1/sapdata2/sr3_1/sr3.data1
channel ORA_DISK_1: datafile checking complete, elapsed time:
00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00005
name=/oracle/RM1/sapdata2/sr3700_1/sr3700.data1
channel ORA_DISK_1: datafile checking complete, elapsed time:
00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00006
name=/oracle/RM1/sapdata2/sr3usr_1/sr3usr.data1
channel ORA_DISK_1: datafile checking complete, elapsed time:
00:00:00
Edit init.ora file /oracle/RM1/112/dbs/init_+DATA.ora. This PFILE
will be used to create the database on the target platform
Run RMAN script /oracle/conv.rman on target platform to convert
datafiles
To recompile all PL/SQL modules, run utlirp.sql and utlpr.sql on
the target platform
To change the internal database identifier, use DBNEWID Utility
```

Finished conversion at source at 21-DEC-10

```

RMAN>
RMAN> **end-of-file**

```

Modify generated script

Note that the script “conv.rman” created in the previous step needs to be modified manually as it may still contain some unwanted tablespaces and datafiles that cannot be transported.

Example: (generated version)

```

$ cat /oracle/conv.rman
STARTUP NOMOUNT PFILE = '/oracle/RM1/112/dbs/init_+DATA.ora';
RUN {
  CONVERT
  FROM PLATFORM 'AIX-Based Systems (64-bit)'
  PARALLELISM 1
  DATAFILE '/oracle/RM1/sapdata1/system_1/system.data1' FORMAT
  '+DATA'
  DATAFILE '/oracle/RM1/sapdata1/sysaux_1/sysaux.data1' FORMAT
  '+DATA'
  DATAFILE '/oracle/RM1/sapdata2/undo_1/undo.data1' FORMAT '+DATA'
  DATAFILE '/oracle/RM1/sapdata2/sr3_1/sr3.data1' FORMAT '+DATA'
  DATAFILE '/oracle/RM1/sapdata2/sr3700_1/sr3700.data1' FORMAT
  '+DATA'
  DATAFILE '/oracle/RM1/sapdata2/sr3usr_1/sr3usr.data1' FORMAT
  '+DATA'
; }

```

Example: (modified version)

```

$ cat /oracle/conv.rman
[oracle@oracx3v3 source_RM1]$ cat conv.rman
STARTUP NOMOUNT;
RUN {
  CONVERT
  FROM PLATFORM 'AIX-Based Systems (64-bit)'
  PARALLELISM 3
  DATAFILE '/oracle/source_RM1/RM1/sapdata2/sr3_1/sr3.data1' FORMAT
  '+DATA'
  DATAFILE '/oracle/source_RM1/RM1/sapdata2/sr3700_1/sr3700.data1'
  FORMAT '+DATA'
  DATAFILE '/oracle/source_RM1/RM1/sapdata2/sr3usr_1/sr3usr.data1'
  FORMAT '+DATA'
; }

```

Note that in this example the path '/oracle' on the source system is NFS mounted under '/oracle/source_RM1' on the target system so that the target system can access the database files. The script 'conv.rman' will also be available on the target system.

Export metadata

Set tablespaces to read only

Before metadata can be exported and tablespaces can be converted all tablespaces need to be set into 'read only' mode.

Example:

```
$ sqlplus
```

```
SQL*Plus: Release 11.2.0.3.0 Production on Tue Dec 21 13:29:45  
2010
```

```
Copyright (c) 1982, 2010, Oracle. All rights reserved.
```

```
Enter user-name: / as sysdba
```

```
Connected to:
```

```
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit  
Production
```

```
With the Partitioning, OLAP, Data Mining and Real Application  
Testing options
```

```
SQL> shutdown immediate;  
Database closed.  
Database dismounted.  
ORACLE instance shut down.  
SQL> startup  
ORACLE instance started.
```

```
Total System Global Area 1219334144 bytes  
Fixed Size 2219088 bytes  
Variable Size 1073742768 bytes  
Database Buffers 134217728 bytes  
Redo Buffers 9154560 bytes  
Database mounted.  
Database opened.
```

```
SQL> alter tablespace psapsr3 read only;
```

```
Tablespace altered.
```

```
SQL> alter tablespace psapsr3700 read only;
```

Tablespace altered.

```
SQL> alter tablespace psapsr3usr read only;
```

Tablespace altered.

Export all metadata

Note that this step exports **all** metadata information (views, synonyms, etc.) except the metadata that belongs to tables in the schema of the SAPSR3 user, statistics and tablespace information. Metadata export of the tables in the SAPSR3 schema is shown in the next chapter.

Create a directory object for 'expdp':

```
SQL> create directory migrate_rm1 as '/oracle';
```

Directory created.

Create a parameter file for 'expdp' and start the export.

Example:

```
> cat ttsexpdict.par
full=y
exclude=table:"in( select table_name from dba_tables where owner
in ('SAPSR3','OPS$RM1ADM'))"
exclude=tablespace
exclude=statistics
directory=migrate_rm1
dumpfile=dict.dmp
logfile=ttsexpdict.log
```

```
> expdp PARFILE=ttsexpdict.par
Export: Release 11.2.0.3.0 - Production on Mon Sep 10 16:15:31
2012
```

Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

Username: / as sysdba

```
Connected to: Oracle Database 11g Enterprise Edition Release
11.2.0.3.0 - 64bit Production
With the Partitioning and Real Application Testing options
Starting "SYS"."SYS_EXPORT_FULL_01": /***** AS SYSDBA
PARFILE=ttsexpdict.par
```

Estimate in progress using BLOCKS method
 Processing object type DATABASE_EXPORT/SCHEMA/TABLE/TABLE_DATA

Export metadata of the tables of user SAPSR3

```
> cat ttsexp.par
directory=migrate_rm1
dumpfile=tbs.dmp
transport_tablespaces=PSAPSR3,PSAPSR3700,PSAPSR3USR
transport_full_check=Y
logfile=ttsexp.log
```

```
> expdp PARFILE=ttsexp.par
```

```
Export: Release 11.2.0.3.0 - Production on Mon Sep 10 16:16:11
2012
Copyright (c) 1982, 2011, Oracle and/or its affiliates. All
rights reserved.
Username: / as sysdba
Connected to: Oracle Database 11g Enterprise Edition Release
11.2.0.3.0 - 64bit Production
With the Partitioning and Real Application Testing options
Starting "SYS"."SYS_EXPORT_TRANSPORTABLE_01": /***** AS
SYSDBA PARFILE=ttsexp.par
.
.
Dump file set for SYS.SYS_EXPORT_TRANSPORTABLE_01 is:
/oracle/tbs.dmp
*****
*****
Datafiles required for transportable tablespace PSAPSR3:
/oracle/RM1/sapdata2/sr3_1/sr3.data1
Datafiles required for transportable tablespace PSAPSR3700:
/oracle/RM1/sapdata2/sr3700_1/sr3700.data1
Datafiles required for transportable tablespace PSAPSR3USR:
/oracle/RM1/sapdata2/sr3usr_1/sr3usr.data1
Job "SYS"."SYS_EXPORT_TRANSPORTABLE_01" successfully completed at
16:16:24
```

```
.
.
```

Please note that the export file is placed under ‘/oracle’ that is NFS mounted to ‘/oracle/source_RM1’ on the target system.

Steps on the target platform

Ensure that you can access and read the scripts generated on the source system as well as the database files and export dump files.

New pfile init<DBSID>.ora

For the next steps (conversion of datafiles, creation of a new database, etc.) a new pfile is required.

The pfile is needed only during the migration and should be replaced by your own pfile or spfile that incorporates all SAP recommended parameters and all Oracle ASM specific parameter changes at the end of migration.

Example:

```
[oracle@oracx3v3 source_RM1]$ cat /oracle/RM1/112/dbs/initRM1.ora
*.control_file_record_keep_time=30
*.db_block_size=8192
*.db_domain='oracledev.wdf.sap.corp'
*.db_files=1500
*.db_name='RM1'
*.filesystemio_options='SETALL'
*.java_pool_size=0
*.log_archive_dest_1='location=+ARCH
valid_for=(ALL_LOGFILES,ALL_ROLES) db_unique_name=RM1'
*.log_archive_dest_state_1='enable'
*.log_archive_format='RM1ARC%t_%s_%r.dbf'
*.log_archive_max_processes=2
*.log_archive_min_succeed_dest=1
*.log_archive_trace=0
*.log_buffer=1048576
*.log_checkpoints_to_alert=TRUE
*.open_cursors=800
*.optimizer_index_caching=50
*.optimizer_index_cost_adj=20
*.os_authent_prefix='ops$'
*.PARALLEL_EXECUTION_MESSAGE_SIZE=16384
*.PARALLEL_MAX_SERVERS=80
*.parallel_min_servers=10
*.PARALLEL_THREADS_PER_CPU=1
*.pga_aggregate_target=1G
*.processes=850
*.QUERY_REWRITE_ENABLED='FALSE'
*.RECYCLEBIN='OFF'
*.remote_login_passwordfile='exclusive'
*.remote_os_authent=true
*.REPLICATION_DEPENDENCY_TRACKING=FALSE
*.sessions=1700
```

```

*.sga_target=1G
*.shared_pool_size=200M
*.star_transformation_enabled='true'
RM1.thread=1
*.timed_statistics=true
*.undo_management='AUTO'
RM1.undo_tablespace='PSAPUNDO'
*.db_create_file_dest='+DATA'
*.db_create_online_log_dest_1='+DATA'
*.db_create_online_log_dest_2='+FLASH'
*.control_files='+DATA/RM1/cntrlRM1.dbf','+ARCH/RM1/cntrlRM1.dbf'
, '+FLASH/RM1/cntrlRM1.dbf'
*.compatible='11.2.0.2.0'
*.DB_RECOVERY_FILE_DEST='+FLASH'
*.DB_RECOVERY_FILE_DEST_SIZE=4G
*._compression_compatibility='11.2.0'
[oracle@oracx3v3 source_RM1]$

```

Convert datafiles to Oracle ASM

Run the RMAN script 'conv.rman' to copy and convert all the datafiles to Oracle ASM.

Example:

```

[oracle@oracx3v3 source_RM1]$ rman target /
Recovery Manager: Release 11.2.0.3.0 - Production on Wed Dec 22
06:52:23 2010

```

Copyright (c) 1982, 2009, Oracle and/or its affiliates. All rights reserved.

connected to target database (not started)

```

RMAN> @conv.rman

```

```

RMAN> STARTUP NOMOUNT;
Oracle instance started

```

```

Total System Global Area      1068937216 bytes
Fixed Size                     2233336 bytes
Variable Size                  264244232 bytes
Database Buffers               796917760 bytes
Redo Buffers                    5541888 bytes

```

```

RMAN> RUN {
2>  CONVERT
3>  FROM PLATFORM 'AIX-Based Systems (64-bit)'
4>  PARALLELISM 3
5>  DATAFILE '/oracle/source_RM1/RM1/sapdata2/sr3_1/sr3.data1'
FORMAT '+DATA'

```

```
6> DATAFILE
'/oracle/source_RM1/RM1/sapdata2/sr3700_1/sr3700.data1' FORMAT
'+DATA'
7> DATAFILE
'/oracle/source_RM1/RM1/sapdata2/sr3usr_1/sr3usr.data1' FORMAT
'+DATA'
8> ; }
Starting conversion at target at 22-DEC-10
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=426 device type=DISK
allocated channel: ORA_DISK_2
channel ORA_DISK_2: SID=856 device type=DISK
allocated channel: ORA_DISK_3
channel ORA_DISK_3: SID=1282 device type=DISK
channel ORA_DISK_1: starting datafile conversion
input file name=/oracle/source_RM1/RM1/sapdata2/sr3_1/sr3.data1
channel ORA_DISK_2: starting datafile conversion
input file
name=/oracle/source_RM1/RM1/sapdata2/sr3700_1/sr3700.data1
channel ORA_DISK_3: starting datafile conversion
input file
name=/oracle/source_RM1/RM1/sapdata2/sr3usr_1/sr3usr.data1
converted datafile=+DATA/rm1/datafile/psapsr3.260.738399155
channel ORA_DISK_1: datafile conversion complete, elapsed time:
00:00:15
converted datafile=+DATA/rm1/datafile/psapsr3700.264.738399155
channel ORA_DISK_2: datafile conversion complete, elapsed time:
00:00:15
converted datafile=+DATA/rm1/datafile/psapsr3usr.256.738399155
channel ORA_DISK_3: datafile conversion complete, elapsed time:
00:00:15
Finished conversion at target at 22-DEC-10
```

Note that each datafile gets a new filename created automatically by Oracle. The names generated follow the “Oracle Managed File” (OMF) naming rules for Oracle ASM:

<TABLESPACE>.<NUMBER>.<NUMBER>

If you have a lot of tablespaces and/or datafiles it is recommended to extract the new datafile names from output above. The names are required during the metadata import!

Create a new database

Before plugging the transported tablespaces into a new database we need to create it. Create a new database on Oracle ASM with just the tablespaces that could not be transported (SYSTEM, SYSAUX, PSAPUNDO, PSAPTEMP, ...).

The newly created database should match your old one in terms of the size of the new tablespaces, the character sets, the size and number of online redo logs, etc.

The script below demonstrates how a new database with UTF8 character set is created.

create database scripts

```
[oracle@oracx3v3 source_RM1]$ cat 1_createdb.sql
connect / as sysdba
shutdown immediate;
startup nomount;
CREATE DATABASE "RM1"
MAXINSTANCES 8
MAXLOGHISTORY 1
MAXLOGFILES 100
MAXLOGMEMBERS 3
MAXDATAFILES 1000
DATAFILE '+DATA' SIZE 700M AUTOEXTEND ON NEXT 10240K MAXSIZE
UNLIMITED EXTENT MANAGEMENT LOCAL
SYSAUX DATAFILE '+DATA' SIZE 600M AUTOEXTEND ON NEXT 10240K
MAXSIZE UNLIMITED
DEFAULT TEMPORARY TABLESPACE "PSAPTEMP" TEMPFILE '+DATA' SIZE
1000M AUTOEXTEND ON NEXT 640K MAXSIZE UNLIMITED
UNDO TABLESPACE "PSAPUNDO" DATAFILE '+DATA' SIZE 200M AUTOEXTEND
ON NEXT 5120K MAXSIZE UNLIMITED
CHARACTER SET UTF8
NATIONAL CHARACTER SET UTF8
LOGFILE GROUP 1 ('+DATA', '+FLASH') SIZE 50M,
GROUP 2 ('+DATA', '+FLASH') SIZE 50M,
GROUP 3 ('+DATA', '+FLASH') SIZE 50M
USER SYS IDENTIFIED BY "secret" USER SYSTEM IDENTIFIED BY
"secret";
```

```
[oracle@oracx3v3 source_RM1]$ cat 2_createcatalog.sql
connect sys/secret as SYSDBA
set echo on
spool 2_createcatalog.log append
@?/rdbms/admin/catalog.sql;
@?/rdbms/admin/catblock.sql;
@?/rdbms/admin/catproc.sql;
@?/rdbms/admin/catoctk.sql;
@?/rdbms/admin/owminst.plb;
```

```
connect system/secret
@?/sqlplus/admin/pupbld.sql;
connect system/secret
set echo on
spool sqlPlusHelp.log append
@?/sqlplus/admin/help/hlpbld.sql helpus.sql;
spool off

[oracle@oracx3v3 source_RM1]$ cat 3_lockaccounts.sql
connect / as sysdba
shutdown immediate;
startup;
SET VERIFY OFF
set echo on
spool lockaccounts.log append
BEGIN
  FOR item IN ( SELECT USERNAME FROM DBA_USERS WHERE
ACCOUNT_STATUS IN ('OPEN', 'LOCKED', 'EXPIRED') AND USERNAME NOT
IN (
'SYS','SYSTEM') )
  LOOP
    dbms_output.put_line('Locking and Expiring: ' ||
item.USERNAME);
    execute immediate 'alter user ' ||
      sys.dbms_assert.enquote_name(
      sys.dbms_assert.schema_name(
      item.USERNAME),false) || ' password expire account lock'
;
  END LOOP;
END;
/
spool off

[oracle@oracx3v3 source_RM1]$ cat 4_postcreate.sql
SET VERIFY OFF
connect sys/secret as sysdba
shutdown immediate;
startup;
set echo on
spool postDBCreation.log append
@?/rdbms/admin/catbundle.sql psu apply;
select 'utl_recomp_begin: ' || to_char(sysdate, 'HH:MI:SS') from
dual;
execute utl_recomp.recomp_serial();
select 'utl_recomp_end: ' || to_char(sysdate, 'HH:MI:SS') from
dual;
shutdown immediate;
connect sys/secret as sysdba
startup mount;
alter database archivelog;
```

```
alter database open;
connect sys/secret as sysdba
set echo on
create spfile='spfileRM1.ora' FROM pfile;
shutdown immediate;
spool off
```

Create Database Roles 'SAPCONN' and 'SAPDBA'

Create 'SAPCONN' and 'SAPDBA' roles by running sapconn_role.sql and sapdba_role.sql from sqlplus.

Refer to SAP notes #834917 and #134592 for more information.

Import all metadata (except tables of schema SAPSR3)

Create a directory object for 'impdp' and run impdp to import all the metadata.

Example:

```
SQL> create directory rmlmigration as '/oracle/source_RM1';
Directory created.

> cat ttsimpdict.par
directory=rmlmigration
dumpfile=dict.dmp
logfile=ttsimpdict.log

.
.

> impdp PARFILE=ttsimpdict.par

Import: Release 11.2.0.3.0 - Production on Tue Sep 11 07:36:56
2012
Copyright (c) 1982, 2011, Oracle and/or its affiliates. All
rights reserved.
Username: / as sysdba

Connected to: Oracle Database 11g Enterprise Edition Release
11.2.0.3.0 - 64bit Production
With the Partitioning and Real Application Testing options
Master table "SYS"."SYS_IMPORT_FULL_01" successfully
loaded/unloaded
Starting "SYS"."SYS_IMPORT_FULL_01":  /***** AS SYSDBA
PARFILE=ttsimpdict.par
Processing object type DATABASE_EXPORT/PROFILE
Processing object type DATABASE_EXPORT/SYS_USER/USER
```

```
Processing object type DATABASE_EXPORT/SCHEMA/USER
.
.
.
```

After this step all the users (e.g. SAPSR3, OPS\$RM1ADM), views, synonyms etc. should have been created in the database. Please note that some of the objects (e.g. views) will be 'invalid' as long as the underlying table is not yet created.

Grant 'SAPCONN' and 'SAPDBA' roles

sapsr3:

```
SQL> grant sapconn, unlimited tablespace to sapsr3;
```

Grant succeeded.

OPS\$<SAPSID>adm:

```
SQL> grant sapdba, unlimited tablespace to ops$rmladm;
```

Grant succeeded.

Import tablespace object definitions

```
> cat ttsimp.par
directory=rmlmigration
dumpfile=tbs.dmp
transport_datafiles='+DATA/rm1/datafile/psapsr3.260.738399155','+
DATA/rm1/datafile/psapsr3700.264.738399155','+DATA/rm1/datafile/p
sapsr3usr.256.738399155'
logfile=ttsimp.log
```

```
> impdp PARFILE=ttsimp.par
```

```
Import: Release 11.2.0.3.0 - Production on Tue Sep 11 07:40:20
2012
```

```
Copyright (c) 1982, 2011, Oracle and/or its affiliates. All
rights reserved.
```

```
Username: / as sysdba
```

```
Connected to: Oracle Database 11g Enterprise Edition Release
11.2.0.3.0 - 64bit Production
```

```
With the Partitioning and Real Application Testing options
Master table "SYS"."SYS_IMPORT_TRANSPORTABLE_01" successfully
loaded/unloaded
```

```
Starting "SYS"."SYS_IMPORT_TRANSPORTABLE_01": /***** AS
SYSDBA PARFILE=ttsimp.par
```

```
Processing object type TRANSPORTABLE_EXPORT/PLUGTS_BLK
```

```
.
```

```
.
.
Job "SYS"."SYS_IMPORT_TRANSPORTABLE_01" successfully completed at
07:40:27
```

Check tablespaces

```
SQL> set linesize 1000
SQL> set pagesize 1000
SQL> select tablespace_name,file_name from dba_data_files order
by tablespace_name,file_name;
```

TABLESPACE_NAME	FILE_NAME
PSAPSR3	+DATA/rm1/datafile/psapsr3.260.738399155
PSAPSR3700	+DATA/rm1/datafile/psapsr3700.264.738399155
PSAPSR3USR	+DATA/rm1/datafile/psapsr3usr.256.738399155
PSAPUNDO	+DATA/rm1/datafile/psapundo.267.738387115
SYSAUX	+DATA/rm1/datafile/sysaux.262.738387107
SYSTEM	+DATA/rm1/datafile/system.263.738387097

```
SQL> select tablespace_name,status from dba_tablespaces;
```

TABLESPACE_NAME	STATUS
SYSTEM	ONLINE
SYSAUX	ONLINE
PSAPUNDO	ONLINE
PSAPTEMP	ONLINE
PSAPSR3	READ ONLY
PSAPSR3700	READ ONLY
PSAPSR3USR	READ ONLY

Switch tablespaces into read/write mode

```
SQL> alter tablespace PSAPSR3 read write;
```

Tablespace altered.

```
SQL> alter tablespace PSAPSR3700 read write;
```

Tablespace altered.

```
SQL> alter tablespace PSAPSR3USR read write;
```

Tablespace altered.

```
SQL> select tablespace_name,status from dba_tablespaces;
```

TABLESPACE_NAME	STATUS
SYSTEM	ONLINE
SYSAUX	ONLINE
PSAPUNDO	ONLINE
PSAPTEMP	ONLINE
PSAPSR3	ONLINE
PSAPSR3700	ONLINE
PSAPSR3USR	ONLINE

Set default tablespace and temporary tablespace

```
SQL> alter user sapsr3 default tablespace psapsr3usr temporary tablespace
psaptemp;
```

User altered.

On some systems the default tablespace of the SAPSR3 user is PSAPSR3. If this is the case on your system change it to PSAPSR3 instead of PSAPSR3USR.

Application of SAP notes

After completing the migration steps above you should refer a number of SAP notes and reapply them to your newly created system.

Note that 'grants' of specific rights or roles to Oracle users are not transported with the TTS approach. This means that if you have re-created 'sapconn' role and the 'sapdba' role by applying the corresponding notes you still need to grant to roles to the users.

If your database contains non-SAP specific objects that require specific grants or roles you will have to assign them manually.

The most important SAP notes are:

SAP note	Comment / SAP note title
1503709	Oracle 11.2.0: Patches / Patch collections for 11.2.0.2
1631931	Oracle 11.2.0: Patches / Patch collections for 11.2.0.3
1519872	SAP Database User Profile SAPUPROF
834917	Oracle Database 10g: New database role SAPCONN (valid for 11gR2)
134592	Importing the SAPDBA role (sapdba_role.sql)
1431793	Oracle 11.2.0: Upgrade Scripts
1508271	Oracle GoldenGate / Triple-O

Checks, spfile, GRID registration

The final steps are the same as with “RMAN duplicate from active database”.

Please refer to chapter Checks, Server Parameter File and Register the database to the GRID.



Moving your SAP Database to Oracle Automatic
Storage Management 11g Release 2

January 2013

Author: Markus Breunig

Contributing Authors: Jan Klokkers, Kurt Broeg,
Andreas Becker

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

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com



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

Copyright © 2013, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

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