

Performing a 32 bit to 64 bit migration using the Transportable Database RMAN feature

This note describes the procedure used to perform a 32 bit to 64 bit conversion of an 11.2.0.3 database on the Linux platform.

The RMAN CONVERT DATABASE command is used to automate the movement of an entire database from one platform (the source platform) to another (the destination platform).

This is provided that the source and destination platforms are of the same endian format.

For example between Linux X86 32 bit and Linux X86 64 bit.

Note the following:

- Certain types of blocks, such as blocks in undo segments, need to be reformatted to ensure compatibility with the destination platform.
- Redo log files and control files from the source database are not transported. New control files and redo log files are created for the new database during the transport process, and an OPEN RESETLOGS is performed once the new database is created
- BFILES are not transported. RMAN provides a list of objects using the BFILE datatype in the output for the CONVERT DATABASE command, but users must copy the BFILES themselves and fix their locations on the destination database.
- Tempfiles belonging to locally managed temporary tablespaces are not transported. The temporary tablespace will be re-created on the target platform when the transport script is run.
- External tables and directories are not transported
- Password files are not transported. If a password file was used with the source database, the output of CONVERT DATABASE includes a list of all usernames and their associated privileges. Create a new password file on the destination database using this information

Pre-Migration Checks

Check if transportable database can be used

Before attempting a platform migration with TDB, verify the target platform is supported for TDB by your source platform. Query the view V\$DB_TRANSPORTABLE_PLATFORM for the target platform name

```
SQL> select platform_name from v$db_transportable_platform where platform_name like 'Linux%';
```

```
PLATFORM_NAME
```

```
-----  
Linux IA (32-bit)  
Linux IA (64-bit)  
Linux x86 64-bit
```

Check for external tables and directories

```
SQL> set serveroutput on  
SQL> declare x boolean;  
begin x := dbms_tdb.check_external; end; 2  
3 /
```

The following directories exist in the database:

```
SYS.APPS_DATA_FILE_DIR, SYS.ECX_UTL_LOG_DIR_OBJ, SYS.ECX_UTL_XSLT_DIR_OBJ,  
SYS.LYCO_AP_REMIT, SYS.DATA_PUMP_DIR, SYS.ORACLE_OCM_CONFIG_DIR
```

```
PL/SQL procedure successfully completed.
```

Directory objects must be created on the target system. Query DBA_DIRECTORIES on the source database to determine the file system locations that must exist on the target system for the directory objects to be usable.

```
SQL> select directory_name,directory_path from dba_directories;
```

```
DIRECTORY_NAME
```

```
-----  
DIRECTORY_PATH
```

```
-----  
DATA_PUMP_DIR
```

```
/oracle/DEV/admin/DEV/dpdump/
```

```
ORACLE_OCM_CONFIG_DIR
```

```
/oracle/DEV/devdb/11.2.0.3/ccr/state
```

```
LYCO_AP_REMIT
/home/filxfer/PROD/ENG/AP_REM
```

```
DIRECTORY_NAME
```

```
-----
DIRECTORY_PATH
```

```
-----
ECX_UTL_XSLT_DIR_OBJ
```

```
/usr/tmp
```

```
APPS_DATA_FILE_DIR
```

```
/oracle/DEV/devdb/11.2.0.3/appsutil/outbound/DEV_oradevint
```

```
ECX_UTL_LOG_DIR_OBJ
```

```
/usr/tmp
```

```
6 rows selected.
```

Ensure that each directory listed in the view DBA_DIRECTORIES points to a valid file system directory, or ASM disk group or directory on the target system.

Identify external table files that will need to be transferred to the target system when indicated in a later step. To identify external table files, run the following query

```
SQL> select directory_path||'/'||location External_file_path from dba_directories a, dba_external_locations b where
a.directory_name=b.directory_name;
```

```
no rows selected
```

Check for BFILES

```
REM
REM List all directories that contain BFILES
REM
set serveroutput on format wrap;
set feedback off;
declare
type cur_type is REF CURSOR;
v_cur cur_type;
v_sqlstmt varchar2(100);
v_bfile_loc bfile;
v_bfile_dir_name varchar2(30);
v_bfile_filename varchar2(250);
```

```

v_bfile_realpath varchar2(4000);
type array_type is table of number index by varchar2(512);
bfile_dirs array_type;
mydir varchar2(512);
total_bfiles number := 0;
begin
-- loop through all columns that are BFILE type
for bf in
(select owner,table_name,column_name
from dba_tab_cols
where data_type='BFILE')
loop
v_sqlstmt:='select '||bf.column_name||' from '
||bf.owner||'. '||bf.table_name;
open v_cur for v_sqlstmt;
loop
fetch v_cur into v_bfile_loc;
exit when v_cur%notfound;
-- get BFILE directory alias and filename
dbms_lob.filegetname(v_bfile_loc, v_bfile_dir_name,
v_bfile_filename);
if bfile_dirs.exists(v_bfile_dir_name) then
bfile_dirs(v_bfile_dir_name) := bfile_dirs(v_bfile_dir_name) + 1;
else
bfile_dirs(v_bfile_dir_name) := 1;
end if;
end loop;
close v_cur;
end loop;
dbms_output.put_line(' ');
dbms_output.put_line('The following directories contain external files for BFILE columns');
dbms_output.put_line('Copy the files within these directories to the same path on the target system');
dbms_output.put_line(' ');
-- loop through array of all directories
mydir := bfile_dirs.first;
while mydir is not null loop
-- resolve the directory alias to a full path
select directory_path
into v_bfile_realpath
from all_directories
where directory_name = mydir;
dbms_output.put_line(v_bfile_realpath);
total_bfiles := total_bfiles + bfile_dirs(mydir);
mydir := bfile_dirs.next(mydir);
end loop;
dbms_output.put_line(' ');
dbms_output.put_line('There are ' || bfile_dirs.count
|| ' directories, ' || total_bfiles
|| ' total BFILEs');
dbms_output.put_line(' ');
end;

```

/

The following directories contain external files for BFILE columns
Copy the files within these directories to the same path on the target system

There are 0 directories, 0 total BFILEs

Export OLAP Analytic Workspaces

Create a directory to save OLAP analytical workspace migration files.

```
mkdir /oracle/stage/aw_migrate
```

As Sys:

Create a directory to save OLAP analytical workspace migration files.

```
SQL> create directory aw_migrate as '/oracle/stage/aw_migrate';
```

Directory created.

Obtain a list of OLAP analytical workspaces to be migrated.

```
SQL> col owner format a15  
col aw_name format a15
```

```
select OWNER, AW_NAME ,PAGESPACES from dba_aws  
where owner != 'SYS' order by 1,2;
```

```
SQL>
```

OWNER	AW_NAME	PAGESPACES
FPA	FPAPJP	308
ZPB	ZPBANNOT	7
ZPB	ZPBCODE	292
ZPB	ZPBDATA	7

For each OLAP analytical workspace from the above query (as per the table below), run the following package procedures to export the workspace.

```
exec dbms_aw.execute( 'aw attach <owner>.<aw_name> rw' );
exec dbms_aw.execute( 'allstat' );
exec dbms_aw.execute( 'export all to eif file ''AW_MIGRATE/<schema>_<aw_name>.eif '' );
exec dbms_aw.execute( 'aw detach <owner>.<aw_name>' );
```

Verify that each export file has been successfully created and copy them all across to the same location on the 64-bit server.

Delete each of the OLAP analytical workspaces.

```
exec dbms_aw.execute( 'aw delete <owner>.<aw_name>' );
```

As Sys:

Execute scripts to remove OLAP from the database.

```
@?/olap/admin/catnoamd.sql
@?/olap/admin/olapidrp.plb
@?/olap/admin/catnoaps.sql
@?/olap/admin/catnoxoq.sql
@?/rdbms/admin/utltp.sql
```

Manually remove additional remaining OLAP database objects.

```
drop procedure sys.xoq_validate;
drop view sys.olap_oladb_reg_attrs_pvt;
drop package body sys.cwm2_olap_installer;

set linesize 120 pagesize 0 feedback off echo off

spool /oracle/stage/aw_migrate/drop_olap_syms.sql

select 'drop public synonym ' || object_name || ';'
  from dba_objects
 where owner = 'PUBLIC'
    and object_type = 'SYNONYM'
    and status <> 'VALID'
    and ( object_name like '%OLAP%' or object_name like '%AW' )
    and object_name not like '%RAW%';

spool off

run script /oracle/stage/aw_migrate/drop_olap_syms.sql

SQL> exec dbms_aw.execute( 'aw attach FPA.FPAPJP rw' );
```

```
SQL> exec dbms_aw.execute( 'allstat' );
SQL> exec dbms_aw.execute( 'export all to eif file 'AW_MIGRATE/FPA_FPAPJP.eif '' );
SQL> exec dbms_aw.execute( 'aw detach FPA.FPAPJP' );
SQL> exec dbms_aw.execute( 'aw attach ZPB.ZPBANNOT rw' );
SQL> exec dbms_aw.execute( 'allstat' );
SQL> exec dbms_aw.execute( 'export all to eif file 'AW_MIGRATE/ZPB_ZPBANNOT.eif '' );
SQL> exec dbms_aw.execute( 'aw detach ZPB.ZPBANNOT' );
SQL> SQL> BEGIN dbms_aw.execute( 'export all to eif file 'AW_MIGRATE/ZPB_ZPBANNOT.eif '' ); END;
```

```
*
ERROR at line 1:
ORA-33390: There are no objects to export.
ORA-06512: at "SYS.DBMS_AW", line 93
ORA-06512: at "SYS.DBMS_AW", line 122
ORA-06512: at line 1
```

```
SQL> exec dbms_aw.execute( 'aw attach ZPB.ZPBCODE rw' );
SQL> exec dbms_aw.execute( 'allstat' );
SQL> exec dbms_aw.execute( 'export all to eif file 'AW_MIGRATE/ZPB_ZPBCODE.eif '' );
SQL> exec dbms_aw.execute( 'aw detach ZPB.ZPBCODE' );
SQL> exec dbms_aw.execute( 'aw attach ZPB.ZPBDATA rw' );
SQL> exec dbms_aw.execute( 'allstat' );
SQL> exec dbms_aw.execute( 'export all to eif file 'AW_MIGRATE/ZPB_ZPBDATA.eif '' );
BEGIN dbms_aw.execute( 'export all to eif file 'AW_MIGRATE/ZPB_ZPBDATA.eif '' ); END;
```

```
*
ERROR at line 1:
ORA-33390: There are no objects to export.
ORA-06512: at "SYS.DBMS_AW", line 93
ORA-06512: at "SYS.DBMS_AW", line 122
ORA-06512: at line 1
```

```
SQL> exec dbms_aw.execute( 'aw detach ZPB.ZPBDATA' );
SQL>exec dbms_aw.execute( 'aw delete FPA.FPAPJP' );
```

```
SQL> exec dbms_aw.execute( 'aw delete ZPB.ZPBCODE' );  
SQL> exec dbms_aw.execute( 'aw delete ZPB.ZPBADATA' );  
SQL> exec dbms_aw.execute( 'aw delete ZPB.ZPBANNOT' );
```

Follow MetaLink Note 352306.1

Prepare the database for Transportable Database

Shut Down the Application

Disconnect users and shutdown all application server processes. Users cannot use any application served by the database until the migration to the new platform is complete.

Shut down and Start the database in READ ONLY mode

TDB requires that the source database be opened in READ ONLY mode. The source database will be unavailable from this step forward.

```
SQL> shutdown immediate;  
SQL> startup mount;  
SQL> alter database open read only;
```

Run the DBMS_TDB.CHECK_DB to Check Database State

Checks to see if:

- Unrecognized target platform name
- Target platform has a different endian format.
- Database is not open read-only
- There are active or in-doubt transactions in the database.
- Database compatibility version is below 10


```

SQL> declare
      retcode boolean;
      begin
      retcode := dbms_tdb.check_db('Linux x86 64-bit)', dbms_tdb.skip_none);
      end;
/
 2      3      4      5      6
PL/SQL procedure successfully completed.
SQL> |

```

CONVERT DATABASE

In this case we are converting the datafiles on the destination source as opposed to converting them on the source host.

The source database (32 bit) data files are stored in a file system while the destination database (64 bit) will be using ASM disk groups instead.

Performing the conversion on the destination system will also avoid any performance overhead on the source system while the conversion process is underway.

We use the **CONVERT DATABASE ON TARGET PLATFORM** RMAN command.

This command will generate a convert script which contains the CONVERT DATAFILE commands to perform the conversion from the source to target format

The transport script contains the actual commands used to recreate the control files on the target database after the data file conversion is completed.

```

RMAN>
CONVERT DATABASE ON TARGET PLATFORM
CONVERT SCRIPT '/tmp/convert_script.rcv'
TRANSPORT SCRIPT '/tmp/transport_script.sql'
FORMAT '+DATA'
db_file_name_convert '/oracle/DEV/devdata/', '+DATA/';
RMAN> 2> 3> 4> 5>

```

In our case we encountered an error because a number of users had been allocated the SYSTEM tablespace as their default tablespace.

```
-----
RMAN-00571: *****
RMAN-00569: ***** ERROR MESSAGE STACK FOLLOWS *****
RMAN-00571: *****
RMAN-03002: failure of conversion at source command at 09/24/2012 20:36:47
ORA-01682: read-only DB cannot allocate temporary space in tablespace SYSTEM

RMAN>
```

```
SQL> 1
1* select username from dba_users where temporary_tablespace='TEMP'
SQL> select username from dba_users where temporary_tablespace='SYSTEM';

USERNAME
-----
JRDSYS
JUTLN
JBSNMP
JRDPLUGINS
FAK
APPS_RO
SYS
SYSTEM
SCOTT
YDDATA
JRDDATA

USERNAME
-----
ANONYMOUS
DIP
XDB
SPATIAL_CSW_ADMIN_USR
ORACLE_OCM
SPATIAL_WFS_ADMIN_USR
APPQOSSYS
XS$NULL
EM_MONITOR
SI_INFORMTN_SCHEMA
AD_MONITOR

22 rows selected.

SQL> |
```

```
SQL> select 'alter user '||username||' default temporary tablespace temp;'
       2 from dba_users where temporary_tablespace='SYSTEM';
```

```
'ALTERUSER' || USERNAME || 'DEFAULTTEMPORARYTABLESPACE TEMP;'
```

```
alter user ORDSYS default temporary tablespace temp;
alter user OUTLN default temporary tablespace temp;
alter user DBSNMP default temporary tablespace temp;
alter user ORDPLUGINS default temporary tablespace temp;
alter user FAK default temporary tablespace temp;
alter user APPS_RO default temporary tablespace temp;
alter user SYS default temporary tablespace temp;
alter user SYSTEM default temporary tablespace temp;
alter user SCOTT default temporary tablespace temp;
alter user MDDATA default temporary tablespace temp;
alter user ORDDATA default temporary tablespace temp;
```

```
'ALTERUSER' || USERNAME || 'DEFAULTTEMPORARYTABLESPACE TEMP;'
```

```
alter user ANONYMOUS default temporary tablespace temp;
alter user DIP default temporary tablespace temp;
alter user XDB default temporary tablespace temp;
alter user SPATIAL_CSW_ADMIN_USR default temporary tablespace temp;
alter user ORACLE_OCM default temporary tablespace temp;
alter user SPATIAL_WFS_ADMIN_USR default temporary tablespace temp;
alter user APPQOSSYS default temporary tablespace temp;
alter user XS$NULL default temporary tablespace temp;
alter user EM_MONITOR default temporary tablespace temp;
alter user SI_INFORMTN_SCHEMA default temporary tablespace temp;
alter user AD_MONITOR default temporary tablespace temp;
```

```
22 rows selected.
```

```
RMAN>
CONVERT DATABASE ON TARGET PLATFORM
CONVERT SCRIPT '/tmp/convert_script.rcv'
TRANSPORT SCRIPT '/tmp/transport_script.sql'
FORMAT '+DATA'
db_file_name_convert '/oracle/DEV/devdata/', '+DATA/';
RMAN> 2> 3> 4> 5>
```

```
Starting conversion at source at 24-SEP-12
using channel ORA_DISK_1
using channel ORA_DISK_2
using channel ORA_DISK_3
using channel ORA_DISK_4
```

```
Directory SYS.APPS_DATA_FILE_DIR found in the database
Directory SYS.ECX_UTL_LOG_DIR_OBJ found in the database
Directory SYS.ECX_UTL_XSLT_DIR_OBJ found in the database
Directory SYS.LYCO_AP_REMIT found in the database
Directory SYS.DATA_PUMP_DIR found in the database
Directory SYS.ORACLE_OCM_CONFIG_DIR found in the database
Directory SYS.AW_MIGRATE 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=00402 name=/oracle/DEV/devdata/a_txn_data03.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00011 name=/oracle/DEV/devdata/a_txn_data04.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00017 name=/oracle/DEV/devdata/undo01.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00023 name=/oracle/DEV/devdata/undo02.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00405 name=/oracle/DEV/devdata/a_txn_ind04.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00393 name=/oracle/DEV/devdata/a_txn_ind01.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00014 name=/oracle/DEV/devdata/a_txn_ind06.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
```

...

.....

.....

```

channel ORA_DISK_1: starting to check datafiles
input datafile file number=00010 name=/oracle/DEV/devdata/olap.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00314 name=/oracle/DEV/devdata/portal01.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00397 name=/oracle/DEV/devdata/a_nolog01.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00019 name=/oracle/DEV/devdata/R11G_discopstore.dbf
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/DEV/devdata/ctxd01.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00007 name=/oracle/DEV/devdata/owad01.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00016 name=/oracle/DEV/devdata/R11G_discoptm5cache.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_1: starting to check datafiles
input datafile file number=00018 name=/oracle/DEV/devdata/R11G_discoptm5meta.dbf
channel ORA_DISK_1: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_2: starting to check datafiles
input datafile file number=00021 name=/oracle/DEV/devdata/a_txn_data05.dbf
channel ORA_DISK_2: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_3: starting to check datafiles
input datafile file number=00392 name=/oracle/DEV/devdata/a_txn_data01.dbf
channel ORA_DISK_3: datafile checking complete, elapsed time: 00:00:00
channel ORA_DISK_4: starting to check datafiles
input datafile file number=00401 name=/oracle/DEV/devdata/a_txn_data02.dbf
channel ORA_DISK_4: datafile checking complete, elapsed time: 00:00:00
Edit init.ora file /oracle/DEV/devdb/11.2.0.3/dbs/init_+DATA.ora. This PFILE will be used to create the database on the target platform
Run RMAN script /tmp/convert_script.rcv on target platform to convert datafiles
Run SQL script /tmp/transport_script.sql on the target platform to create database
To recompile all PL/SQL modules, run utlirp.sql and utlirp.sql on the target platform
To change the internal database identifier, use DBNEWID Utility
Finished conversion at source at 24-SEP-12

RMAN> █

```

Copy the /tmp/convert_script.rcv and /tmp/transport_script.sql files from source to appropriate destination on target machine.

Copy init.ora parameter file from source to target \$ORACLE_HOME/dbs

Make changes to init.ora

Change control_files location to point to ASM disk group

Control_files='+DATA','+DATA','+DATA'

Also change location of adump in the init.ora datafile.

Mount the source (read only) file system on target server

Add an entry in /etc/exports

```
$ cat /etc/exports
/oracle oradevdb.mycorp.com.au (ro,sync)
```

Followed by

```
/etc/init.d/nfs restart
```

On target

```
mount -t nfs 192.168.xxx.xx:/oracle /oracle
```

Run the CONVERT script

convert_script.rcv contents:

```
RUN {
  CONVERT
  FROM PLATFORM 'Linux IA (32-bit)'
  PARALLELISM 4
  DATAFILE '/oracle/DEV/devdata/undo02.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/undo01.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/undo04.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/undo03.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/system06.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/system07.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/system11.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/system01.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/system02.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/system03.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/system04.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/system05.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/system10.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/system09.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/system08.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/a_txn_data03.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/a_txn_data04.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/a_txn_ind04.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/a_txn_ind01.dbf' FORMAT '+DATA'
  DATAFILE '/oracle/DEV/devdata/a_txn_ind06.dbf' FORMAT '+DATA'
```

```

DATAFILE '/oracle/DEV/devdata/a_txn_ind02.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_txn_ind03.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_txn_ind05.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/sysaux01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_archive01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_media01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_int01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_txn_data06.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_ref01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_summ01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_ref02.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/lyco_data01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/xxpic01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_txn_data07.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_queue01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_queue02.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/discoverer01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/odm.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/olap.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/portal01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_nolog01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/R11G_discopstore.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/ctxd01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/owad01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/R11G_discoptm5cache.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/R11G_discoptm5meta.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_txn_data05.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_txn_data01.dbf' FORMAT '+DATA'
DATAFILE '/oracle/DEV/devdata/a_txn_data02.dbf' FORMAT '+DATA'
; }

```

Next, start the database with NOMOUNT option and run the RMAN script convert_script.rcv

```

[oradev@oradevdb dbs]$ export ORACLE_SID=DEV
[oradev@oradevdb dbs]$ sqlplus sys as sysdba

SQL*Plus: Release 11.2.0.3.0 Production on Mon Sep 24 21:46:24 2012

Copyright (c) 1982, 2011, Oracle. All rights reserved.

Enter password:
Connected to an idle instance.

SQL> startup nomount;
ORACLE instance started.

Total System Global Area 1068937216 bytes
Fixed Size 2235208 bytes
Variable Size 432014520 bytes
Database Buffers 620756992 bytes
Redo Buffers 13930496 bytes
SQL>

```

RMAN>@convert_script.rcv

```
channel ORA_DISK_4: datafile conversion complete, elapsed time: 00:00:40
channel ORA_DISK_4: starting datafile conversion
input file name=/oracle/DEV/devdata/a_nolog01.dbf
converted datafile=+DATA/a_queue01.dbf
channel ORA_DISK_1: datafile conversion complete, elapsed time: 00:01:51
channel ORA_DISK_1: starting datafile conversion
input file name=/oracle/DEV/devdata/R11G_discopstore.dbf
converted datafile=+DATA/r11g_discopstore.dbf
channel ORA_DISK_1: datafile conversion complete, elapsed time: 00:00:07
channel ORA_DISK_1: starting datafile conversion
input file name=/oracle/DEV/devdata/ctxd01.dbf
converted datafile=+DATA/olap.dbf
channel ORA_DISK_2: datafile conversion complete, elapsed time: 00:00:11
channel ORA_DISK_2: starting datafile conversion
input file name=/oracle/DEV/devdata/owad01.dbf
converted datafile=+DATA/portal01.dbf
channel ORA_DISK_3: datafile conversion complete, elapsed time: 00:00:11
channel ORA_DISK_3: starting datafile conversion
input file name=/oracle/DEV/devdata/R11G_discoptm5cache.dbf
converted datafile=+DATA/a_nolog01.dbf
channel ORA_DISK_4: datafile conversion complete, elapsed time: 00:00:11
channel ORA_DISK_4: starting datafile conversion
input file name=/oracle/DEV/devdata/R11G_discoptm5meta.dbf
converted datafile=+DATA/ctxd01.dbf
channel ORA_DISK_1: datafile conversion complete, elapsed time: 00:00:02
converted datafile=+DATA/owad01.dbf
channel ORA_DISK_2: datafile conversion complete, elapsed time: 00:00:02
converted datafile=+DATA/r11g_discoptm5cache.dbf
channel ORA_DISK_3: datafile conversion complete, elapsed time: 00:00:01
converted datafile=+DATA/r11g_discoptm5meta.dbf
channel ORA_DISK_4: datafile conversion complete, elapsed time: 00:00:01
Finished conversion at target at 25-SEP-12
```

RMAN> **end-of-file**

RMAN> shutdown immediate;

Oracle instance shut down

Note – what I found that while it mentions the converted datafile as “+DATA/olap.dbf”, actually the file is created as “+DATA/dev/datafile/olap.277.794979681”

We had to take lines out of transport_script.sql and edit to include the correct ASM OMF file name.

We then created a file called crectl.sql which had the following contents:

crectl.sql

```
CREATE CONTROLFILE REUSE SET DATABASE "DEV" RESETLOGS FORCE LOGGING ARCHIVELOG
  MAXLOGFILES 32
  MAXLOGMEMBERS 5
  MAXDATAFILES 512
  MAXINSTANCES 8
  MAXLOGHISTORY 7260
LOGFILE
GROUP 1 (
  '+DATA',
  '+DATA'
) SIZE 100M BLOCKSIZE 512,
GROUP 2 (
  '+DATA',
  '+DATA'
) SIZE 100M BLOCKSIZE 512,
GROUP 3 (
  '+DATA',
  '+DATA'
) SIZE 100M BLOCKSIZE 512,
GROUP 4 (
  '+DATA',
  '+DATA'
) SIZE 100M BLOCKSIZE 512,
GROUP 5 (
  '+DATA',
  '+DATA'
) SIZE 100M BLOCKSIZE 512,
GROUP 6 (
  '+DATA',
  '+DATA'
) SIZE 100M BLOCKSIZE 512,
GROUP 7 (
  '+DATA',
  '+DATA'
) SIZE 100M BLOCKSIZE 512,
GROUP 8 (
  '+DATA',
  '+DATA'
) SIZE 100M BLOCKSIZE 512,
GROUP 9 (
  '+DATA',
```

```
'+DATA'
) SIZE 100M BLOCKSIZE 512,
GROUP 10 (
'+DATA',
'+DATA'
) SIZE 100M BLOCKSIZE 512
DATAFILE
'+DATA/dev/datafile/apps_ts_tx_data.355.794965469',
'+DATA/dev/datafile/apps_ts_tx_data.361.794965469',
'+DATA/dev/datafile/apps_ts_tx_data.360.794965469',
'+DATA/dev/datafile/apps_ts_tx_data.356.794965469',
'+DATA/dev/datafile/apps_ts_tx_idx.351.794969911',
'+DATA/dev/datafile/apps_undots1.353.794969911',
'+DATA/dev/datafile/apps_undots1.352.794969911',
'+DATA/dev/datafile/apps_ts_tx_data.354.794969895',
'+DATA/dev/datafile/apps_ts_tx_idx.350.794973207',
'+DATA/dev/datafile/apps_ts_tx_idx.348.794973643',
'+DATA/dev/datafile/apps_ts_tx_idx.349.794973643',
'+DATA/dev/datafile/apps_ts_tx_idx.347.794973797',
'+DATA/dev/datafile/sysaux.343.794976769',
'+DATA/dev/datafile/apps_undots1.344.794976663',
'+DATA/dev/datafile/apps_undots1.345.794976619',
'+DATA/dev/datafile/system.342.794977855',
'+DATA/dev/datafile/apps_ts_tx_data.337.794978951',
'+DATA/dev/datafile/apps_ts_seed.336.794979017',
'+DATA/dev/datafile/apps_ts_summary.335.794979051',
'+DATA/dev/datafile/apps_ts_seed.334.794979059',
'+DATA/dev/datafile/lyco.316.794979229',
'+DATA/dev/datafile/system.320.794979215',
'+DATA/dev/datafile/xxpic.315.794979255',
'+DATA/dev/datafile/apps_ts_tx_data.319.794979261',
'+DATA/dev/datafile/system.318.794979377',
'+DATA/dev/datafile/system.317.794979377',
'+DATA/dev/datafile/system.307.794979377',
'+DATA/dev/datafile/system.268.794979377',
'+DATA/dev/datafile/system.271.794979523',
'+DATA/dev/datafile/system.267.794979523',
'+DATA/dev/datafile/system.269.794979523',
'+DATA/dev/datafile/apps_ts_queues.270.794979523',
'+DATA/dev/datafile/odm.276.794979665',
'+DATA/dev/datafile/olap.277.794979681',
'+DATA/dev/datafile/portal.278.794979695',
'+DATA/dev/datafile/discoverer.275.794979665',
'+DATA/dev/datafile/apps_ts_queues.273.794979665',
'+DATA/dev/datafile/rllg_disco_pstore.281.794979715',
'+DATA/dev/datafile/apps_ts_nologging.279.794979711',
'+DATA/dev/datafile/owapub.283.794979731',
'+DATA/dev/datafile/rllg_disco_ptm5_cache.284.794979731',
'+DATA/dev/datafile/ctxd.282.794979729',
'+DATA/dev/datafile/rllg_disco_ptm5_meta.285.794979731',
'+DATA/dev/datafile/system.272.794979657'
```

```
CHARACTER SET US7ASCII  
;
```

Now startup nomount the database and run the script crectl.sql which will create the control files in the ASM disk group +DATA

After the control files have been created we open the database with RESETLOGS option.

```
SQL> ALTER DATABASE OPEN RESETLOGS;
```

The generated transport_script.sql has the commands to add the tempfiles to the database and then run utlirp.sql.

But we found that when we tried to add the tempfiles, we were getting an error as shown below.

```
SQL> ALTER TABLESPACE TEMP ADD TEMPFILE '+DATA'  
      SIZE 9000M AUTOEXTEND ON NEXT 8192 MAXSIZE 15000M; 2  
ALTER TABLESPACE TEMP ADD TEMPFILE '+DATA'  
*  
ERROR at line 1:  
ORA-00604: error occurred at recursive SQL level 1  
ORA-06553: PLS-801: internal error [56327]
```

Next shutdown the database using the IMMEDIATE option.

Change the location for CONTROL_FILES from +DATA to actual OMF file name

```
*.control_files='+DATA/DEV/CONTROLFILE/current.323.795010677','+DATA/dev/controlfile/current.322.795010679','+DATA/dev/controlfile/current.321.795010679'
```

Then start the database using the UPGRADE option

```
SQL> startup upgrade pfile=' /u01/DEV/db/tech_st/11.2.0.3/dbs/initDEV.ora'
```

Run the utlirp.sql script

```
@ ?/rdbms/admin/utlirp.sql
```

Add the tempfiles to the database

```
ALTER TABLESPACE TEMP ADD TEMPFILE '+DATA'  
    SIZE 9000M AUTOEXTEND ON NEXT 8192 MAXSIZE 15000M;  
ALTER TABLESPACE TEMP ADD TEMPFILE '+DATA'  
    SIZE 2000M AUTOEXTEND ON NEXT 8192 MAXSIZE 15000M;  
ALTER TABLESPACE TEMP ADD TEMPFILE '+DATA'  
    SIZE 3200M AUTOEXTEND ON NEXT 8192 MAXSIZE 32000M;  
ALTER TABLESPACE R11G_IAS_TEMP ADD TEMPFILE '+DATA'  
    SIZE 104857600 AUTOEXTEND ON NEXT 8192 MAXSIZE 1048576000 ;
```

Shutdown the database

```
SQL> shutdown immediate
```

Next, start the database and run utlrp.sql to recompile all PL/SQL modules

```
SQL> startup pfile='/u01/DEV/db/tech_st/11.2.0.3/dbs/initDEV.ora'
```

```
SQL>@ ?/rdbms/admin/utlrp.sql
```

Note – recompilation took close to 3 hours in this case

Add OLAP back into the database.

```
@?/olap/admin/olap.sql SYSAUX TEMP
```

Import OLAP Analytic Workspaces

For each OLAP analytical workspace , run the following package procedures to import the workspace.

```
exec dbms_aw.execute( 'aw create <owner>.<aw_name>' );  
exec dbms_aw.execute( 'import all from eif file "AW_MIGRATE/<schema>_<aw_name>.eif" data dfns' );  
exec dbms_aw.execute( 'update' );  
commit;  
exec dbms_aw.execute( 'aw detach <owner>.<aw_name>' );
```

Drop the directory used to import the OLAP analytical workspace migration files.

```
SQL> drop directory aw_migrate;
```

Recompile invalid objects.

```
SQL> @?/rdbms/admin/utlrlp.sql
```

```
SQL> select OWNER, AW_NAME ,PAGESPACES from dba_aws  
       where owner != 'SYS' order by 1,2;
```

OWNER	AW_NAME	PAGESPACES
FPA	FPAPJP	532
ZPB	ZPBANNOT	8
ZPB	ZPBCODE	480
ZPB	ZPBDATA	8

Note – further details can be found in MetaLink Note 352306.1