**Trademarks**

The following terms are registered trademarks of International Business Machines Corporation in the United States and/or other countries: AIX, AIX/L, AIX/L (AIX/L®), IBM, IBM (IBM®), pSeries, Total Storage, Power PC.

The following terms are registered trademarks of International Business Machines Corporation in the United States and/or other countries; Power VM, Advanced Micro-Partitioning, AIX 5L, AIX 6 (logo), Micro Partitioning, Power Architecture, POWER6, POWER6, POWER7, Redbooks, System p, System p5, System p6, System p7, System Storage.

A complete list of U.S trademarks owned by IBM may be found at:

Oracle, Java and all Java based trademarks are registered trademarks of Oracle Corporation in the USA and/or other countries

UNIX is a registered trademark in the United States, other countries or both.

**Disclaimers**

While efforts have been made to verify the information, this paper may contain errors. IBM makes no warranties or representation with respect to the content hereof and specifically disclaim any implied warranties of merchantability or fitness for any particular purpose. IBM assumes no responsibility for any errors that may appear in this document. The information contained in this document is subject to change without notice. IBM reserves the right to make any such changes without obligation to notify any person of such revisions or changes. IBM makes no commitment to keep the information contained herein up to date.
Table of Contents

Introduction .................................................................................................................. 5

Oracle Transportable Database .................................................................................. 6

Environment ................................................................................................................ 7

Prior to Migration ....................................................................................................... 8

*Hardware / Server* ................................................................................................... 8
  Initial AIX 6.1 tuning ............................................................................................... 8

*Software* .................................................................................................................. 8
  AIX............................................................................................................................ 8
  Oracle ....................................................................................................................... 9
  a) rootpre.sh ............................................................................................................ 9
  b) Oracle Universal Installer (OUI) ......................................................................... 9
  c) Other known OUI issues & workaround ............................................................ 9
  d) Oracle patches for AIX ...................................................................................... 9
  e) Other Oracle patches ......................................................................................... 10

*Storage & Other Considerations* ............................................................................ 10
  a) JFS2 mount option for transfer of files. ............................................................ 10
  b) Data file fragmentation ..................................................................................... 11
  c) Identifying & fixing fragmentation issues ......................................................... 11

*Pre Migration Tasks (on the source database server)* .......................................... 13

*Shutdown source database* .................................................................................... 13

*Open Database in READ ONLY mode* ................................................................. 14

*Verify Portability* .................................................................................................. 14

*Check for external objects* .................................................................................... 14

*Identify data files for conversion* .......................................................................... 15
SQL statement for creating control file ........................................ 15
Edit script .................................................................................... 15
Shutdown source database .......................................................... 16

Database Migration (on Target Database Server) ...................... 16
Transfer Data files from the source database server ............... 16
Create Control file ....................................................................... 16
Convert data files using RMAN .................................................. 16
Tuning the conversion of data files ........................................... 17
Shutdown the database ............................................................... 18
Copy files from staging area ....................................................... 18

Open Database & miscellaneous completion tasks ............... 18
 a) Create Temporary Table space ......................................... 19
 b) External Objects .................................................................. 19
 c) Compile Invalid Objects ..................................................... 19
 d) Setup and Start the Database listener ................................. 19

Summary ....................................................................................... 20

References & Acknowledgements .............................................. 21
Introduction

Oracle database operations are similar on all operating systems. However deployment strategies can vary on different operating systems. Migration of an existing Oracle database to a new architecture can prove to be overwhelming initially due to many factors like

- New operating system
- Operating system preparedness
- Installation nuances
- Oracle patches
- Database migration

This document is targeted towards users migrating from Solaris and HP-UX operating systems to AIX and covers topics mentioned above for Oracle database version 10gR2 and higher. It also provides a step by step migration example using transportable databases, talks about tuning a migration and avoiding and overcoming migration pitfalls.

The migration steps documented in this paper have been leveraged at numerous customer sites successfully and lessons learnt during these migrations are also included.

For the purpose of this document, the source environment is on Sun Solaris and will be referred to as source database. The AIX environment will be referred to as the target database.
Oracle Transportable Database

Oracle Database 8i introduced the Transportable Table Space (TTS) feature. However, with the release of Oracle 10gR1, this feature has become more useful because the transportable table space feature now supports cross-platform migration.

With the release of Oracle Database 10g Release 2 (10gR2), there is now support for transporting an entire database across platforms that share the same endianness. (Note: Endianness refers to the method for sequencing byte order. The big-endian convention arranges most significant byte first, little-endian sequences less significant byte first.). This feature is now known as Transportable Database (TDB).

In short, transportable table space has evolved into Transportable Database in Oracle 10gR2 and higher.

This paper assumes that the reader has:

- A high degree of Oracle database-administrator (DBA) skills and is, therefore, well-versed with the import (imp) and export (exp) process
- A fair amount of UNIX® system-administrator skills or the assistance of a UNIX system administrator who can help you in moving files from one system to another

By transporting the entire database, it is now possible to replicate the database environment across platforms that share the same endianness. The following table classifies the big-endian and little-endian platforms.

<table>
<thead>
<tr>
<th>Big-endian</th>
<th>Little-endian</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX and Linux on POWER™</td>
<td>Microsoft® Windows® (32- and 64-bit bit)</td>
</tr>
<tr>
<td>IBM System z™ and Linux</td>
<td>Windows 64-bit for AMD®</td>
</tr>
<tr>
<td>HP® UX (IA-64)</td>
<td>HP Tru64 UNIX</td>
</tr>
<tr>
<td>HP UX (64-bit)</td>
<td>HP Open VMS</td>
</tr>
<tr>
<td>Sun™ Solaris™ (SPARC 32- and 64-bit)</td>
<td>Solaris (x86)</td>
</tr>
<tr>
<td>Apple® Mac OS</td>
<td>Linux IA (32/64 bit), Linux 64-bit for AMD</td>
</tr>
</tbody>
</table>

Oracle10gR2 (10.2.0.1) or higher is required for transporting a database. The next few steps explain the process for transporting an Oracle 10gR2 database from Sun Solaris to AIX 6. These steps are the same as they would be if you were transporting a database from any big-endian platform to AIX.
Environment

Source System Configuration:
Hardware: Sun Fire V440 Server
- 4 x Ultra SPARC IIIi @ 1.59 GHz
- 32 GB RAM
- Direct Attach Storage D1000 Arrays.

Software:
- Solaris 5.9
- Oracle 10gR2 (10.2.0.4) or higher

Target System Configuration:
Hardware: IBM p6 570
- 2 x POWER6 CPU @ 3.5 GHz
- 32 GB RAM
- IBM N3700.

Software:
- AIX 6
- Visual Age C/C++ Professional v7.0 (only required for Pro*C, otherwise runtime is sufficient)
- Oracle 10gR2 (10.2.0.4) or higher
Prior to Migration

**Hardware / Server**

**Initial AIX 6.1 tuning**

a) Network tuning (interface level)
   - tcp_sendspace=262144
   - tcp_recvspace=262144
   - rfc1323=1

b) ioo
   - numfsbufs=1568
   - j2_maxPageReadAhead=256

c) Misc
   - ulimit –a for oracle user should return unlimited for all (except core)
   - maxuproc=16384
   - export AIXTHREAD_SCOPE=S for oracle user

For generic AIX tuning, please refer to a document titled ‘**Oracle Architecture & Tuning on AIX**’ and a link to this document is included in the reference section

**Software**

**AIX**

Following file sets are required for a successful Oracle Enterprise Server installation. (Single instance only)

- bos.adt.base
- bos.adt.lib
- bos.perf.libperfstat
- bos.perf.perfstat
- bos.perf.protocols
- xlC.aix.rte

Use the command ‘lslpp –l | grep <file set name> to verify the existence of a file set. AIX patches should be consistent with My Oracle Support (MOS) note id. 282036.1 – Minimum software version and patches required to support Oracle products on IBM Power System..
**Oracle**
The Oracle database version and patches should be consistent with the source server.

**a) rootpre.sh**
rootpre.sh script is required to be run as a root user prior to invoking the Oracle Universal Installer (OUI). Depending on the Oracle & AIX versions, rootpre.sh prepares AIX system for Oracle installation. This script should be executed only once for multiple oracle homes/versions. Do not run the rootpre.sh if there is a newer release of Oracle database already installed on the system.

**b) Oracle Universal Installer (OUI)**
Oracle installer (OUI) in 10gR2 does not recognize AIX 6.1 as a supported operating system. Use the `runInstaller –ignoreSysPrereqs`.

**c) Other known OUI issues & workaround**
- OUI crash with IBM JDK/JRE 1.5 on POWER7 architecture – Ref. MOS Note 1065579.1
- OUI has detected processes running – Ref. MOS Note 287428.1

**d) Oracle patches for AIX**
Patches listed in this section are bare minimum and pertain only to migration and/or performance. These patches must be installed to begin a successful transition to AIX.

1. **Oracle Release 10.2.0.4 (10gR2)**
   - Patch 7226548 – Default enable 64k pagesize support for SGA
   - Patch 7568734 – Sporadic spikes of ‘log file sync’ on AIX with heavy commit concurrency
   - Patch 6784747 – Wasted memory in koh-kghu allocations.

2. **Oracle Release 10.2.0.5**
   - Patch 7226548 – Default enable 64k pagesize support for SGA (The two additional patches mentioned in the 10.2.0.4 section are included in the 10.2.0.5 patchset)

3. **Oracle Release 11.1.0.7 (11gR1)**
   - Patch 7893016 – RMAN convert fails with ORA-600

4. **Oracle Release 11.2.0.1(11gR2)**
   - Patch 7893016 – RMAN convert fails with ORA-600
Oracle 11gR2 (11.2.0.2 and 11.2.0.3) require additional performance related Oracle patches. Please refer to the link for “IBM Power7 AIX and Oracle Database Performance considerations” in the reference section. This document provides a list of additional Oracle patches that may be required

e) Other Oracle patches

The source database server may have some additional generic Oracle patches installed, and for consistency these patches should also be installed on the target database server.

On the source database server do an ‘opatch lsinventory’ to generate a list of Oracle patches installed. Download the equivalent AIX patches from Oracle support site and install it.

After the patch installation, create the init.ora file, and test it out with a ‘startup nomount’ command.

Storage & Other Considerations

Bulk of Oracle database migration time is spent on copying data files between two different host systems. File transfer is accomplished by copying via NFS mounted file systems or FTP over the network.

Before the transfer of data files, it’s important to discuss a couple of issues which can cause performance degradation and/or slow the copy process.

a) JFS2 mount option for transfer of files.

For the purpose of copy/ftp, do not explicitly mount the file system with ‘cio’ option as this will considerably slow down the copy process because file system operations with a small block size are less efficient on a ‘cio’ mounted file system.

For the purpose of copying, It is recommended to mount the target file system with release behind write (rbw) option. This will enhance serial copy process and also releases the memory pages after they are written to disk. As soon as the copy process is completed, unmount and mount the file system with default options.

For general database operations for Oracle 10g & higher on AIX, it is recommended to use concurrent I/O (cio) via Oracle (init.ora parameters filesystemio_options=setall & disk_async_io=true), with default mount options and the control files and redo logs should be isolated in a separate JFS2 file system which has been created with a block size (agblksize) of 512 bytes.
Explicit cio mount option should only be used if there is a need for an external access (cp, dd, cpio) to the oracle data files after the data base has been opened.

It is beyond the scope of this document to discuss all other AIX/Oracle tuning topics, however this is covered in a document titled ‘Oracle Architecture & Tuning on AIX’ and a link to this document is included in the reference section.

b) Data file fragmentation.

Fragmentation will occur when multiple copy/ftp processes are run concurrently on a single file system to copy data files in parallel. Data file fragmentation has the potential to introduce serious performance issues, hence it important to do it right the first time.

In order to avoid fragmentation, it is recommended to run the copy/ftp process sequentially within each file system. You have the flexibility to copy multiple file systems in parallel while observing the recommended serial copy procedure within each file system.

c) Identifying & fixing fragmentation issues.

AIX provides the fileplace command which displays the placement of a specified file within the logical or physical volume containing the file.

fileplace –v displays more information about the file and its placement, including statistics on how widely the file is spread across the volume and the degree of fragmentation in the volume.

```
#fileplace -v tx_idx09.dbf
File: tx_idx19.dbf  Size: 2097160192 bytes  Vol: /dev/oradata_lv
Blk Size: 4096  Frag Size: 4096  Nfrags: 512002
Inode: 73913  Mode: -rw-r-----  Owner: oracle  Group: dba

Logical Extent
------------------
40870240-40870271             32 frags       131072 Bytes,   0.0%
40870368-40870399             32 frags       131072 Bytes,   0.0%
40870464-40870495             32 frags       131072 Bytes,   0.0%
40870560-40870591             32 frags       131072 Bytes,   0.0%
40870656-40870687             32 frags       131072 Bytes,   0.0%
...
41217280-41217311             32 frags       131072 Bytes,   0.0%
41217344-41637599             420256 frags   1721368576 Bytes,  82.1%
01625266-01625267             2 frags         8192 Bytes,   0.0%

512002 frags over space of 40012334 frags:  space efficiency = 1.3%
2845 extents out of 512002 possible:  sequentiality = 99.4%
```
The sequentiality % is more important than the space efficiency %. A high sequentiality % indicates that when a file is read, it won't be moving the actuator (a seek) much due to fragmentation. A high space efficiency indicates the file's pieces are all close together, but the file could be fragmented requiring lots of seeks to read it, but at least the actuator won't have to move far.

The only way to defragment a fragmented file is to create a scratch or temporary file system and sequentially copy out the file and sequentially copy in again.
Pre Migration Tasks (on the source database server)

Disable archive log mode and shutdown source database

If archive log mode is enabled on the source database, edit spfile/pfile to set the initialization parameter log_archive_start = false. The source database should be shutdown in the manner described below using the following sequence of shutdown commands

- shutdown immediate
- startup mount
- alter database noarchivelog; (Use only if archive log mode is enabled)
- alter database open read only;
- shutdown normal

Using the above steps for shutdown ensures that archive log mode is disabled and pending changes within the database are synced with the data files.
**Open Database in READ ONLY mode**

In this section we prepare the source database for migration. The first step is to open the database in READ ONLY mode as follows.

```sql
SQL> startup mount
ORACLE instance started.
Total System Global Area 4815060992 bytes
Fixed Size                  2077752 bytes
Variable Size             905972680 bytes
Database Buffers         3892314112 bytes
Redo Buffers               14696448 bytes
Database mounted.
SQL> alter database open read only;
Database altered.
SQL>
```

**Verify Portability**

The next step is to verify the portability of the database by running the following SQL script which uses the DMS_TDB.CHECK_DB function. If the database is not transportable the SQL script will give an error message.

```plsql
SET SERVEROUTPUT ON
DECLARE
    db_ready BOOLEAN;
BEGIN
    db_ready := DBMS_TDB.CHECK_DB('AIX-Based Systems (64-bit)');
END;
/
PL/SQL procedure successfully completed.
```

**Check for external objects**

The next step checks for the existence of any external objects within the database like directories, tables and lists them.

```plsql
SET SERVEROUTPUT ON
DECLARE
    external BOOLEAN;
BEGIN
    external := DBMS_TDB.CHECK_EXTERNAL;
END;
/
The following directories exist in the database:
SYS.APPS_DATA_FILE_DIR, SYS.HZ_DNB_SOURCE_DIR, SYS.DATA_PUMP_DIR
```
**Identify data files for conversion**

The next step identifies the ‘SYSTEM’ and ‘UNDO’ table space data files. These file will be converted using the RMAN tool on the target server. Use the following SQL statement to identify the SYSTEM and UNDO data files in your environment.

```
SQL> select FILE_NAME "Datafiles requiring Conversion" from DBA_DATA_FILES
2   where TABLESPACE_NAME in (select distinct TABLESPACE_NAME
3   from DBA_ROLLBACK_SEGS);
```

<table>
<thead>
<tr>
<th>Datafiles requiring Conversion</th>
</tr>
</thead>
<tbody>
<tr>
<td>/oracle/oracle/system01.dbf</td>
</tr>
<tr>
<td>/oracle/oracle/undo.dbf</td>
</tr>
</tbody>
</table>

2 rows selected.

**SQL statement for creating control file**

The next step generates the trace file which contains the ‘create control file’ statement for control file. The file is generated in the oracle user dump (udump) area as defined in the init.ora.

The file will be used on the target database server to create the control file.

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

**Edit script**

Edit the trace file which was generated earlier to create the control file. The trace generates two SQL statements for the creating the control file. For the purpose of migration we will use the section of RESETLOGS (usually set #2 in the file). Edit the path to reflect the locations of the data files on the target server. Delete all the other lines.

The contents of your file would be similar as follows

```
STARTUP NOMOUNT PFILE='/oracle/db/10gr2/dbs/initPROD.ora'
CREATE CONTROLFILE REUSE SET DATABASE "VIS" RESETLOGS NOARCHIVELOG
   MAXLOGFILES 32
   MAXLOGMEMBERS 5
   MAXDATAFILES 512
   MAXINSTANCES 8
   MAXLOGHISTORY 3635
LOGFILE
   GROUP 1 '/oracle/oradata/log1.dbf' SIZE 7999M,
   GROUP 2 '/oracle/oradata/log2.dbf' SIZE 7999M
DATAFILE
   '/oracle/oradata/sys1.dbf',
   .
   .
   '/oracle/oradata/apps_assm_idx.dbf'
CHARACTER SET UTF8;
spool off
```

After editing, copy/transfer this script to the target server for later use
**Shutdown source database**

The source database can be shutdown (shutdown immediate) or left in a read-only mode until the migration completes.

The next section describes the step to migrate the database on the target server.

**Database Migration (on Target Database Server)**

The action now shifts to the Target Database server.

**Transfer Data files from the source database server**

The first step involves transferring the Oracle data files from the source database server using NFS mount or via FTP. Ensure that the files are placed in the file system as mentioned in the create control file script. Redo logs and temp table space data files need not be transferred as they will be re-created on the target database server..

**Create Control file**

In the second step the control file is created. The init.ora parameter file should be set up at this point as per the pre-requisites. Verify the location of the control files in the init.ora. Run the script to create the control file.

**Convert data files using RMAN**

In the third step, SYSTEM & UNDO table spaces related data files are converted using the RMAN command. *The other data files are not converted* (Ref. MOS note id 732053.1 – Avoid data file conversion during Transportable Database)

```
$ORACLE_HOME/bin/rman target / log='convert.log'
@@convert_script.rman

. Starting backup at 01-FEB-07
  . input filename=/oracle/oradata/system01.dbf
  converted datafile=/oradata_new/system01.dbf
  channel ORA_DISK_1: datafile conversion complete, elapsed time: 00:01:16
  Finished backup at 01-FEB-07

Starting backup at 01-FEB-07
  using channel ORA_DISK_1
  using channel ORA_DISK_2
  channel ORA_DISK_1: starting datafile conversion
  input filename=/oracle/undo.dbf
  converted datafile=/oradata_new/undo.dbf
  channel ORA_DISK_1: datafile conversion complete, elapsed time: 00:01:06
  Finished backup at 01-FEB-07

RMAN>
RMAN> **end-of-file**
```
Tuning the conversion of data files

In every migration exercise, the migration window is usually of limited time duration; hence time is of the essence. With some creativity we can convert data files in parallel and reduce the overall conversion time to as much as fifty percent. This can be achieved as follows

- create one script for every data file to be migrated
- create a shell script which calls the individual RMAN script and runs it in the background mode

```
$ cat datafile1.rman
RUN {
    CONVERT DATAFILE '/oracle/oradata/system01.dbf'
      FROM PLATFORM 'Solaris[tm] OE (64-bit)'
      FORMAT '/oradata_new/system01.dbf';
}
EXIT;

$ cat datafile2.rman
RUN {
    CONVERT DATAFILE '/oracle/oradata/system02.dbf'
      FROM PLATFORM 'Solaris[tm] OE (64-bit)'
      FORMAT '/oradata_new/system02.dbf';
}
EXIT;

$ cat parallel_convert.sh
#!/bin/ksh
rman target=/ @datafile1.rman log='datafile1.log' &
rman target=/ @datafile2.rman log='datafile2.log' &
rman target=/ @datafile3.rman log='datafile3.log' &
wait
$rman target=/ @datafile4.rman log='datafile4.log' &
```

In the above example the datafile1.rman and datafile2.rman represent scripts for individual files to be converted. The shell script parallel_convert.sh starts the RMAN command in background mode and waits until the third data file conversion is completed before it starts another set of data file conversions. Depending on the CPU utilization the wait statement can be put after two or four data files. In this way we can convert multiple data files.
**Shutdown the database**

Examine the conversion log files for any error. After the conversion is completed successfully, the database is shutdown.

```sql
SQL> shutdown immediate
ORA-01109: database not open
Database dismounted
ORACLE instance shutdown
```

**Copy files from staging area**

Once the conversion is completed, we copy the converted data files (SYSTEM & UNDO) from the staging area to the location specified in the control file. In this example the data files are moved from staging area (/oradata_new) to /oracle/oradata. As per the create control file statements, the data files should be present in the /oracle/oradata directory.

```bash
$ cd /oradata_new
$ pwd
/oradata_new
$ cp * /oracle/oradata
$ 
```

**Open Database & miscellaneous completion tasks**

Lastly the database is opened; a recovery is performed using the backup control file and redo logs are reset as follows.

```sql
SQL> startup mount
ORACLE instance started.
Total System Global Area 1744830464 bytes
Fixed Size 2071768 bytes
Variable Size 654312232 bytes
Database Buffers 1073741824 bytes
Redo Buffers 14704640 bytes
Database mounted.
SQL> recover database using BACKUP CONTROLFILE until cancel;
.
.
.
Specify log: {=suggested | filename | AUTO | CANCEL} cancel ........................................ Type Cancel
Media recovery cancelled.
SQL> alter database open resetlogs;
```

The database migration is complete, however before the database can be used, there are a few tasks which need to be completed.
a) **Create Temporary Table space**

Create the temporary table space as sized on the source database server

```
SQL> create temporary tablespace temp_table_space_name
SQL> tempfile '/xxx/temp.dbf' size 32m autoextend on
SQL> extent management local;
```

b) **External Objects**

Use export / import, Data Pump or recreate the external objects on target database server.

c) **Compile Invalid Objects**

Login with SQLPLUS as sysdba and follow the sequence of commands to compile the invalid objects

```
SQL> shutdown immediate;
SQL> startup upgrade;
SQL> @ ?/rdbms/admin/utlrp.sql
SQL> shutdown immediate;
SQL> startup;
SQL> @ ?/rdbms/admin/utlrp
```

**Note.** The amount of time required to recompile the database object is dependent on the size and number of objects within the database.

d) **Setup and Start the Database listener**

Copy the listener.ora from the source database server and make necessary editing to reflect the target hostname and the oracle home. Start the database listener.

The database migration is now complete. It’s advisable to take a cold backup at this time.
Summary

Proper planning is the key for Oracle platform migration projects. It’s important to line up all resources prior to execution. Multiple mock migrations are usually the key to tuning migrations.

Oracle Transportable Database certainly helps in migrating complex databases. The introduction of Transportable Database provides a highly performing and straightforward migration technology than earlier Export/Import or Data Pump options. However some infrastructure challenges remain as an obstacle for faster cross platform migration. Transferring large data files from source to target via the network can take time. This can be challenging for very large database sizes which can run into multiple terrabytes. Technologies like VERITAS Portable Data Containers can be utilized to overcome moving the data files.

This concludes the topics for this document. Please refer to the listings in the “References” section for sources of additional information on questions that you may have,
References & Acknowledgements

My Oracle Support (formerly Metalink)

Oracle Architecture & Tuning on AIX
http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP100883

IBM Power7 AIX and Oracle Database Performance considerations
http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102171

My Oracle Support (MOS) Notes
• 282036.1 - Minimum software versions and patches required to support Oracle products on IBM pSeries
• 468472.1 – rootpre.sh script and rootpre directory are missing from the Oracle 11g Release 1
• 1401921.1 – Cross-Platform Database Migration (across same endian) using RMAN Transportable Database

Oracle patches
• Patch 6613550 – rootpre.sh

Acknowledgements

Several people provided technical input and guidance for this paper including:

• Peter Barnett
• Dan Braden
• Timothy Spencer
• Stephen Poon