Performing Cross-Platform Migration using Oracle Recovery Manager for Oracle Database 12c - lessons learned

Cross-Platform Oracle RMAN restore was introduced in Oracle Database 12c, this document describes the lessons learned from testing this functionality on IBM Systems.

Andrew Martin Braid - andrew.braid@fr.ibm.com
IBM Oracle Center Montpellier
ioc@fr.ibm.com
July 2017
Performing Cross-Platform Migration using Oracle Recovery Manager for Oracle Database 12c - lessons learned
http://www.ibm.com/support/techdocs
© Copyright 2017, IBM Corporation
Abstract

This document describes the Cross-Platform Migration method using Oracle Recovery Manager (RMAN) for Oracle Database 12c. This is a new feature in Oracle Database 12c. The intended audience for this document is a database administrator or a person with some knowledge of Oracle Database and Oracle RMAN.

Introduction

Cross-Platform RMAN Restore is a new feature of Oracle Database 12c which allows an Oracle RMAN backup to be restored on a platform other than that on which the original backup was performed.

This gives us an additional migration method which is faster than traditional methods such as datapump.

The document is based on tests that were performed as a proof of concept for a customer migration.

This document outlines a number of tests that were performed and the many errors and issues that were encountered and how we corrected them or worked around them.

Notice

This document is based on testing performed by the IBM Oracle Center and the experience of the IBM Oracle Center team using the Oracle product functionality.

THIS DOCUMENT IS NOT OFFICIAL IBM OR ORACLE DOCUMENTATION.

This document is presented “As-Is” and IBM does not assume responsibility for the statements expressed herein.

The information contained in this paper resulted from:

Oracle and IBM documentation
Experience gained from tests run in the IBM Oracle Center Montpellier
Questions on the capabilities of non-IBM products should be addressed to the suppliers of these products.

Prerequisites

The reader should have knowledge of the following:

- Oracle RMAN
- Oracle sqlplus
Overview of the testing performed

The IBM Oracle Center received a request from a client to test Cross-Platform Migration using Oracle RMAN incremental backups. The idea was to perform an initial level 0 backup which would be sent by courier (the database was 17TB in size) with an incremental level 1 backup sent via FTP for the changes that occurred during the restore of the level 0 backup.

A preliminary test plan was established to validate this new Oracle product functionality which only became available with Oracle Database version 12c.

The test plan was as follows:

- Create a database on Linux® running on an IBM z Systems® server
- Backup the database running on Linux using the Oracle RMAN cross-platform method with incremental level 0
- Transfer the backup to an IBM Power® Systems server with POWER8™ processors
- Restore the database on the Power Systems™ server
- Make a change to the database running on Linux on the z Systems server to allow a new backup to be performed
- Backup the database running on Linux on the z Systems server using the Oracle RMAN Cross-Platform Method with incremental level 1
- Transfer the level 1 backup to the Power Systems server
- Restore the level 0 and level 1 backups on the Power Systems server

Pre-requisites

Before you create a cross-platform backup to transport a database across platforms, the following prerequisites must be met:

- The COMPATIBLE parameter in the server parameter file of the source database and the destination database must be set to 12.0.0 or higher.
- The source database must be open in read-only mode.
- The source platform and destination platform must use the same endian format. For Power Systems servers and Linux on z Systems servers the following platforms are compatible:

<table>
<thead>
<tr>
<th>PLATFORM_ID</th>
<th>PLATFORM_NAME</th>
<th>ENDIAN_FORMAT</th>
<th>CON_ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Solaris® OE (32-bit)</td>
<td>Big</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Solaris® OE (64-bit)</td>
<td>Big</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>AIX-Based Systems (64-bit)</td>
<td>Big</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>HP-UX (64-bit)</td>
<td>Big</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>HP-UX IA (64-bit)</td>
<td>Big</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>IBM zSeries Based Linux</td>
<td>Big</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>Apple Mac OS</td>
<td>Big</td>
<td>0</td>
</tr>
<tr>
<td>18</td>
<td>IBM Power Based Linux</td>
<td>Big</td>
<td>0</td>
</tr>
</tbody>
</table>

- The DBMS_TDB.CHECK_DB procedure must run successfully.
**Running DBMS_TDB.CHECK_DB**

This package is launched with the following command in SQLPlus. The platform value passed is that of the target database platform. To run the code the database must be started READ ONLY.

```sql
SET SERVEROUTPUT ON
DECLARE
db_ready BOOLEAN;
BEGIN
    db_ready := DBMS_TDB.CHECK_DB('AIX-Based Systems (64-bit)');
END;
/
```

The script returns:

*PL/SQL procedure successfully completed.*

If no warnings appear, or if `DBMS_TDB.CHECK_DB` returns `TRUE`, then you can transport the database.

**Overview of the migration method**

Once the pre-requisites had been performed the following steps were required to complete the migration.

1. Restart DB as read only
2. RMAN backup
3. Backup Controlfile to backup area
4. Copy pfile(nit ora) to backup area
5. Copy pwd file (if used) to backup area
6. Copy backup area to new server
7. Copy pfile(nit ora) to ORACLE_HOME/dbs
8. Create directories for database files
9. Start db nomount
10. RMAN restore
11. Rebuild Controlfile from script
12. Open database

**Perform an Oracle RMAN backup for transport**

To be able to perform a restore on another platform the backup must be performed with the database open in READ ONLY mode. This means that the RMAN backup is disruptive and means that the database is not available for users to continue working.

The cross-platform backup will not be part of the existing backup schedule because cross-platform backups are not recorded in Oracle RMAN or controlfile.
There are three syntaxes possible for the backup.

A cross-platform backup can only be used on another platform, if you try to restore the backup onto the same platform it will fail with the following error:

*RMAN-05093: Restore cannot be mixed with cross-platform datafile restore*

### Syntax

In each of the syntax examples the same format has been used. It has been defined with the configuration parameter:

```
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT '/u01/oracle/db12c102/oradata/backup_test/%U';
```

If this parameter is not used the format of the backup destination must be defined on the command line.

The three syntaxes for performing a cross-platform backup are as follows:

#### For transport

```sql
backup for transport database;
```

FOR TRANSPORT tells Oracle RMAN that this backup is done for a cross-platform restore. It can be restored on any platform other than the one where the backup was performed.

#### To platform

For Linux on a z Systems or LinuxOne server:

```sql
backup to platform 'IBM zSeries Based Linux' database;
```

For AIX running on a Power Systems server:

```sql
backup to platform 'AIX-Based Systems (64-bit)' database;
```

The TO PLATFORM option means that the database can only be restored on that particular platform.

#### Implicit

```sql
BACKUP DATABASE;
```

Because the database is open as READ ONLY the backup is implicitly created as a cross-platform backup.

### Oracle RMAN backup options

The following parameters were tested.
Incremental level 0

If the “for transport” or “to platform” syntax is used for the backup command all incremental backups will be performed as level 0 backups which means a full database backup. As the backup is not part of the regular cycle of backups it is not recorded in either the controlfile or the Oracle RMAN catalog. This means that there is no reference point for an incremental backup to be based upon.

Incremental level 1

If an incremental level 1 backup is run using the “for transport” or “to platform” syntax a level 0 backup or full backup is performed, it is not possible to perform an incremental level 1 backup despite the command completing successfully with either the “for transport” or “to platform” syntaxes.

If the incremental level 0 backup was performed implicitly, without the “for transport” or “to platform” syntax, a level 1 backup is performed.

Unfortunately, when restoring this backup an ORA-600 error occurs when opening the database. An SR was opened for this error but Oracle Support said: “You will not be able to use incremental backup with for transport option”. They suggested using the method described in the Oracle Support document: 12C - Reduce Transportable Tablespace Downtime using Cross Platform Incremental Backup (Doc ID 2005729.1) but this solution only works on the Linux x86-64 platform.

Section Size

Section size is not compatible with the FOR TRANSPORT option. It will generate the following error:

RMAN-08408: Cannot specify SECBYTES clause when producing cross-platform backups

The section size can be used with the implicit backup command but the restore of a backup with section size was not part of the tests performed for the customer. Using section size will result in a list of backup set pieces which will need to be restored individually.

Parallelism

In the examples given in this document we have not defined individual channels. Many DBAs still allocate channels individually but in the testing performed for the customer, the backup was done with the configuration parameter for parallelism. This can be set as in the example below:

```
CONFIGURE DEVICE TYPE disk PARALLELISM 4;
```

The value of the parallelism should be defined based on the CPU count on the smaller system. Oracle suggests that the value should be set to two times the CPU count.
Example of performing the backup

Restart the database as read only

Run the following commands in sqlplus against a sys as sysdba:

```
Shutdown immediate;
startup mount;
alter database open read only;
exit;
```

If you do not open the database as read only the backup command will fail with:

`RMAN-06920: database <ORACLE_SID> is not open read-only`

Connect to rman

As the owner of the Oracle binaries, with the ORACLE_SID parameter set correctly, launch rman at the unix command prompt. The rman command should be in your PATH, it is located in `$ORACLE_HOME/bin`.

```
rman
Connect target /
```

Perform the backup

You can perform the backup using the following command:

```
BACKUP as compressed backupset
FOR TRANSPORT
FORMAT '<backup directory>/%U'
DATABASE;
```

The compressed option is not mandatory but it is compatible with cross-platform RMAN.

Replace `<backup directory>` with the directory location where you want to save the backup to or remove the format clause and set this in your configuration parameters as seen above.

Configuration file backup

If you are using an spfile create a pfile from your spfile. Copy the pfile or init.ora to the backup directory. Copy the orapw file to the backup directory.

Create controlfile script

The controlfiles are not included in the backup, you will need to recreate them on the new platform.

You can create the script for the controlfile creation using a command such as:

```
alter database backup controlfile to trace '/tmp/backup_ctl.txt';
```
This will put the output into a text file at the path of your choosing, in this example: ‘/tmp/backup_ctl.txt’.

You need to choose the second section of the output in this file which gives the command for resetting the redologs which are not part of the backup.

**Sample output**

In this example, the database is an empty database created by dbca. We have specified the FORMAT in the command but this could equally have been defined in the configuration parameters of Oracle RMAN as described at the start of the syntax section of this document.

```sql
BACKUP
FOR TRANSPORT
FORMAT '/u01/oracle/db12c102/oradata/backup_example/%U'
DATABASE;2> 3> 4>

Starting backup at 12-06-2017 14:44:53
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=416 device type=DISK
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00003
name=/u01/oracle/db12c102/oradata/TEST12/sysaux01.dbf
input datafile file number=00001
name=/u01/oracle/db12c102/oradata/TEST12/system01.dbf
input datafile file number=00004
name=/u01/oracle/db12c102/oradata/TEST12/undotbs01.dbf
input datafile file number=00006
name=/u01/oracle/db12c102/oradata/TEST12/users01.dbf
channel ORA_DISK_1: starting piece 1 at 12-06-2017 14:44:54
channel ORA_DISK_1: finished piece 1 at 12-06-2017 14:45:09
piece handle=/u01/oracle/db12c102/oradata/backup_example/01s6k7j6_1_1
tag=TAG20170612T144454 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:15
Finished backup at 12-06-2017 14:45:09
```

This backup has created just one backup set piece, 01s6k7j6_1_1, in the directory specified.

**Restoring a cross-platform backup**

Copy the following from the source to the target server and set the correct permissions for:

- `pfile` or `init.ora` to `$ORACLE_HOME/dbs`
- `oraenv` file (if used) to `$ORACLE_HOME/dbs`
- script from the backup controlfile command
- `rman` backupset pieces

Ensure that the directories have been created for the following files and that the ownership and permissions have been set correctly:

- controlfiles (this location is in the `pfile` which may need to be updated)
Performing Cross-Platform Migration using Oracle Recovery Manager for Oracle Database 12c - lessons learned

- redologs (this location is in the controlfile creation script)
- datafiles (this location is in the controlfile creation script)

**Syntax**

Unlike the backup, there is only one syntax for the restore but we cannot set the configuration parameters at the RMAN level as the database is not mounted. This means that we need to pass the format for the new datafiles on the command line. The restore does not automatically restore the files to the same location. If we run the command without the formatting or "to new" options we will get the following error:

```
RMAN-00558: error encountered while parsing input commands
RMAN-01009: syntax error: found "from": expecting one of: "format, to"
RMAN-01007: at line 1 column 26 file: standard input
```

There are many options that can be used in the restore command which are listed below:

**from platform**

This parameter is optional, the restore works with or without it:

```
restore from platform 'IBM zSeries Based Linux' foreign database ....
```

**To new**

The "to new" option is used in conjunction with OMF (Oracle Managed Files).

This means that the parameter DB_CREATE_FILE_DEST needs to be set or the restore command will fail with the error:

```
RMAN-05088: DB_CREATE_FILE_DEST is not set
```

For Oracle Database 12c OMF is activated by default in databases created using DBCA. OMF will automatically recreate a new filename for each datafile.

**Format**

The datafile names can be formatted using this option, this is explained in more detail in the example below.

**Example of restore**

In this example, the "to new" option has been used and the database is configured to use OMF which is the default for a database created by DBCA with Oracle Database 12c.

As the owner of the oracle binaries:
Performing Cross-Platform Migration using Oracle Recovery Manager for Oracle Database 12c - lessons learned
http://www.ibm.com/support/techdocs
© Copyright 2017, IBM Corporation

The names of all of the datafiles have been changed during the restore process by OMF and will be different at each restore. This means that the controlfile script needs to be updated every time because the filenames will be different. Clearly this increases the risk of human error if the cross platform restore is part of a migration with a database with lots of datafiles.

**Set newname**

When restoring a normal RMAN backup the following syntax would set the new datafile name to the same datafile name used on the source database but in the new location:

```
SET NEWNAME FOR DATABASE TO '/bckp/test_data/TEST12/datafile/%b';
```
Using the `%b` option is not possible with a cross platform restore, this could be another bug as the documentation doesn't detail the options but this option does not work in the format string.

A set newname command would have to be used to reset each individual filename, for example:

```
SET NEWNAME FOR DATAFILE 1 TO '%...';
```

Performing the newname file by file could be very long for a database with hundreds of files.

**Format**

The alternative to the “to new” option is to provide the format string. For this we need to turn off OMF.

We can turn off OMF with the following command:

```
alter system set '_omf' = disabled scope=spfile;
```

Note that the database needs to be restarted, the parameter cannot be changed with `scope=memory` or `scope=both`.

If OMF is turned off then the format string must be used. Using the syntax “restore foreign database to new” will result in RMAN hanging with the following error written to the alert log:

```
Here is an example of the command output:

```
RMAN> restore from platform 'IBM zSeries Based Linux' foreign database format '/bckp/test_data/TEST12/datafile/%N%f.dbf'
   from backupset '/bckp/testrman/backup_example/01s6k7j6_1_1';
2>
Starting restore at 12-JUN-17
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=242 device type=DISK

channel ORA_DISK_1: starting datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup set
channel ORA_DISK_1: restoring all foreign files in backup piece
channel ORA_DISK_1: reading from backup piece
/bckp/testrman/backup_example/01s6k7j6_1_1
channel ORA_DISK_1: restoring foreign file 3 to
/bckp/test_data/TEST12/datafile/SYSAUX3.dbf
channel ORA_DISK_1: restoring foreign file 1 to
/bckp/test_data/TEST12/datafile/SYSTEM1.dbf
channel ORA_DISK_1: restoring foreign file 4 to
/bckp/test_data/TEST12/datafile/UNDOTBS14.dbf
channel ORA_DISK_1: restoring foreign file 6 to
/bckp/test_data/TEST12/datafile/USERS6.dbf
channel ORA_DISK_1: foreign piece
handle=/bckp/testrman/backup_example/01s6k7j6_1_1
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:26
Finished restore at 12-JUN-17
```

The datafile names are not the same as those created by DBCA but they are unique (due to the use of the file ID) and they will be the same after each restore so your controlfile creation script will not need to be recreated each time.

**Restoring a parallel backup**

A backup that has been performed in parallel will have several backup set pieces. Oracle suggests that we can restore them with a command such as the following:

```
restore from platform 'IBM zSeries Based Linux' foreign database format '/bckp/test_data/TEST12/datafile/%N%f.dbf'
   from backupset '/bckp/testrman/backup20/0drsns4g_1_1','/bckp/testrman/backup20/0ersns4g_1_1','/bckp/testrman/backup20/0brsns4g_1_1','/bckp/testrman/backup20/0crsns4g_1_1';
```

In reality, Oracle RMAN only processes the first file and ignores any subsequent files.

The restore command needs to be performed for each backup set piece in turn for example:

```
RMAN> restore from platform 'IBM zSeries Based Linux' foreign database format '/bckp/test_data/TEST12/datafile/%N%f.dbf'
   from backupset '/bckp/testrman/backup20/0crsns4g_1_1';

restore from platform 'IBM zSeries Based Linux' foreign database format '/bckp/test_data/TEST12/datafile/%N%f.dbf'
   from backupset '/bckp/testrman/backup20/0brsns4g_1_1';

...
Creating the new controlfile and open the database

Once the database has been restored you will need to create the controlfile.

You need to choose the second command in the file that was created by the backup controlfile to trace command which has the “resetlogs” option.

You will need to update the directory and filenames for the datafiles and redologs.

Once this command has been run you will be ready to open the database with the resetlogs option.

Sample output

```
CREATE CONTROLFILE SET DATABASE "TEST12" RESETLOGS ARCHIVELOG
   MAXLOGFILES 16
   MAXLOGMEMBERS 3
   MAXDATAFILES 100
   MAXINSTANCES 8
   MAXLOGHISTORY 292
LOGFILE
   GROUP 1 ( '/u01/oracle/db12c102/oradata/TEST12/onlinelog/redo01.log' )
          SIZE 50M BLOCKSIZE 512,
   GROUP 2 ( '/u01/oracle/db12c102/oradata/TEST12/onlinelog/redo02.log' )
          SIZE 50M BLOCKSIZE 512,
   GROUP 3 ( '/u01/oracle/db12c102/oradata/TEST12/onlinelog/redo03.log' )
          SIZE 50M BLOCKSIZE 512
-- STANDBY LOGFILE
DATAFILE
   '/bckp/test_data/TEST12/datafile/SYSTEM1.dbf',
   '/bckp/test_data/TEST12/datafile/SYSAUX3.dbf',
   '/bckp/test_data/TEST12/datafile/UNDOTBS14.dbf',
   '/bckp/test_data/TEST12/datafile/USERS6.dbf'
CHARACTER SET WE8MSWIN1252
2> 3> 4> 5> 6> 7> 8> 9> 10> 11> 12> 13> 14> 15> 16> 17> 18> 19> 20> 21> 22> 23> 24> ;

using target database control file instead of recovery catalog
Statement processed

RMAN> alter database open resetlogs;
Statement processed
```
Summary

Before Oracle Database 12c, Oracle RMAN backup was only possible on the same platform. Cross-platform RMAN restore allows us to back up a database on one platform and restore it on another.

For a cross-platform RMAN migration the backup must be performed with the database open READ ONLY. This means that the backup is disruptive as the database must be shut down and restarted in READ ONLY mode.

Because the database does not need to rebuild the tables and indexes individually this method is faster than an export and import using datapump.

Unfortunately, incremental backups cannot be used for cross-platform RMAN restore, the backups must be full or incremental level 0 which is effectively a full backup.

Advantages of this solution

RMAN backup is a faster solution than datapump export/import for the recreation of a database. This document does not provide performance tests comparing the two methods but Oracle RMAN is more efficient than datapump by design and this is particularly true for the restore.

Because the Oracle RMAN backup contains the whole database we can be confident that all the objects, indexes, packages, procedures, views, grants etc. have been migrated. We don’t need to check the data, rebuild indexes, validate the referential integrity or recalculate segment statistics after the migration.

Disadvantages of this solution

The number of platforms for which this solution is possible is relatively low due to the limitation on endianness but it remains a list of high end servers which IBM are commonly in competition with and will potentially want to migrate from.

This is a disruptive method as the database has to be restarted in READ ONLY mode before the start of the migration. This is not a zero or near zero downtime migration solution.
Resources

These Web sites provide useful references to supplement the information contained in this document:

- Database Backup and Recovery User's Guide - Chapter 28 - Transporting Data Across Platforms
  https://docs.oracle.com/database/121/BRADV/rcmxplat.htm#BRADV05432

- Oracle RMAN in Oracle Database 12c: New Features and Best Practices

About the author

Andrew Braid started working with Oracle products in 1996 for World Wide Support in the UK. Originally specializing in Oracle E-Business Suite as part of the Core Technologies group. He spent many years working as an independent consultant on Oracle E-Business Suite migration projects and as a production DBA at a number of high profile clients. Andrew joined IBM in 2011 to provide Oracle product support for the benchmarking teams at the IBM Client Center in Montpellier.
Performing Cross-Platform Migration using Oracle Recovery Manager for Oracle Database 12c - lessons learned
http://www.ibm.com/support/techdocs
© Copyright 2017, IBM Corporation

15
Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.