

Oracle Technical Migration Procedure – Power to PowerVS Version 1.0

AIX version

March 29, 2024

Michael Wind
Mark Hays
Ralf Schmidt-Dannert

Table of Contents

DOCUMENT INFORMATION	3
VERSION CONTROL.....	3
TARGET AUDIENCE AND INTENT	4
SCOPE AND COVERAGE	4
DISCLAIMER.....	4
DISCOVERY AND MIGRATION OPTION SELECTION	5
DISCOVERY PROCESS	5
MIGRATION OPTIONS COVERED BY THIS DOCUMENT.....	5
SELECTION CONSIDERATIONS	6
TECHNICAL MIGRATION PROCEDURE – BACKUP AND RESTORE APPROACH	7
ASSUMPTIONS.....	7
PREPARATION	8
RMAN BACKUPS.....	9
<i>Assumptions</i>	9
<i>Considerations</i>	10
<i>Network Transfer – Standard Protocols vs. Aspera</i>	10
<i>Procedure</i>	11
<i>Option 1: RMAN Duplicate Database</i>	11
<i>Option 2: RMAN Restore/Recover Database</i>	16
RMAN RESTORE	22
<i>Two Restore Options</i>	22
<i>Assumptions</i>	23
<i>Connectivity and Preparation – Oracle on PowerVS</i>	24
<i>Procedure</i>	24
<i>Option 1: RMAN Duplicate Database</i>	28
<i>Option 2: RMAN Restore/Recover Database</i>	30
APPENDIX	33
SEAGATE LYVE MOBILE SOLUTION FOR PHYSICAL DATA TRANSFER	33
<i>Overview</i>	33
<i>Product and Service Brochure:</i>	33
<i>Physical Devices and Associated Capacities:</i>	33

Document Information

Version Control

Release History	Date	Comments
1.0	March 29, 2024	Initial release covering two RMAN backup and restore migration options. A third option that incorporates Oracle Data Guard in the migration will be included in a subsequent version.

Target Audience and Intent

This documentation presents Oracle Database Administrators (DBAs) with two options, both based on Oracle RMAN, to migrate AIX-based Oracle databases from IBM Power to Power Virtual Server (PowerVS).

The target audience consists of solution and infrastructure architects and Oracle database administrators.

The intent is to present representative steps to execute plans and procedures to perform Oracle database migrations, recognizing that each migration scenario presents unique challenges in terms of deployment, configuration and available resources.

Scope and Coverage

The following procedures assume availability of sufficient network connectivity to support data transfer and/or data replication as required.

- A discussion of alternative procedures using the Seagate Lyve Mobile Solution to transfer On-Premises backups/database files to PowerVS infrastructure is interspersed. Links to detailed information about this service may be found in the document appendix.

Task steps required will vary per implementation. Refer to cited Oracle documentation for details regarding the execution of specific commands.

Disclaimer

Any attempt to execute these procedures will be performed in context with Customer's established procedures for operating and maintaining non-production and/or production systems. Customer will take customary actions to ensure system availability for maintenance and/or reconfiguration as required, and schedule downtime as required. Customer is responsible for reviewing these representative procedures in the context of their particular environment and adjusting as required.

The Oracle Database migration options described are not necessarily specific to PowerVS migrations. Oracle technical staff should recognize the procedures used and understand that not every technical detail or consideration have been explicitly identified. The person(s) executing the procedures are expected to understand the full scope of Oracle database backup and recovery methods – including those details not explicitly stated.

Discovery and Migration Option Selection

Discovery Process

This document assumes that a detailed discovery has previously collected:

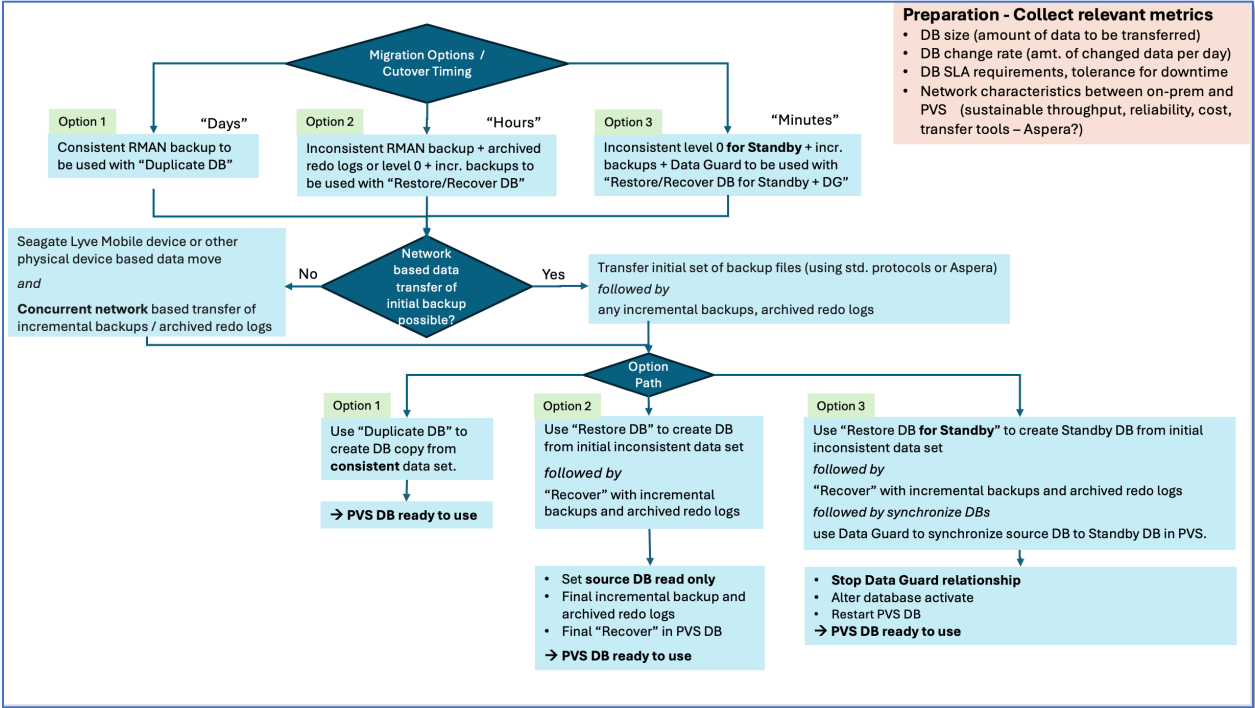
- business, technical, functional, non-functional requirements of the Oracle database being migrated (including access and availability requirements)
- installation configuration, HW/SW inventory and sizing information related to the source database and the underlying platform.
 - Collect relevant metrics related to sizing, change rate, service level requirements, tolerance for migration downtime, capacity of networks to support data transfer from source to target, etc.
- the target environment has been evaluated and deployed consistent with source system capacity/performance/availability as found in the discovery process.

Migration Options Covered by this Document

The following flowchart illustrates the two options currently covered by this document

- **Option 1:** Consistent RMAN backup from source, with generated files to be used by the RMAN Duplicate Database procedure to restore to PowerVS.
- **Option 2:** Inconsistent RMAN backup + archived redo logs (or level 0 + incremental backup(s)) from source, with backup files to be used by the RMAN Restore/Recover Database procedure to restore to PowerVS.

and a third option, leveraging Oracle Data Guard, which will be covered in a document update.



Selection Considerations

Customer requirements and technical conditions affect the decision to select from the presented options.

- 1) Service Level Requirements, particularly when it comes to Restore Point Objectives and Restore Time Objectives.
 - a. Option 1 as presented assumes that the Customer is comfortable with a considerable service delay (RTO = “Days”) before the source database is migrated to the target database. This expectation is often the case for non-production workloads but rarely for production databases.
 - b. Option 2 as presented assumes that the Customer is comfortable with a service interruption in terms of “Hours”. This expectation is often the case for non-production workloads and some production databases.
 - c. Option 3 (not yet presented in this document), incorporating Oracle Data Guard database synchronization, features a potential cutover of service from source to target in minutes.
- 2) Database Size, Network Throughput and Reliability
 - a. Transporting backups of a large database across a network will take time. Consider the following transfer example of 600 GB of backups across an end-to-end network connection where available protocols and bandwidth support 80 megabytes/second throughput. At this rate, it will take 125 minutes to transfer the data. If the database backups are much bigger or if the network connection throughput is less, more transfer time will be required. Unreliable networks may disrupt the transfer, requiring you to restart the transfer process, incurring delay.
 - i. Customers can benefit from specialized transfer tools that compress data before/during transfer, use high-throughput protocols and decompress data after transfer.
 - ii. Access to IBM’s Aspera, which can greatly accelerate data transfers from on-premise to PowerVS locations is very advantageous.
- 3) Availability of Skills to Execute.
 - a. Options provided in this document require experienced DBA skills and the ability to work with infrastructure teams to migrate database content to a new target database on Power Virtual Server within a PowerVS workspace.

Any migration procedures a Customer elects to use should be executed in the context of a detailed, well-rehearsed transfer and cutover plan.

Technical Migration Procedure – Backup and Restore Approach

The RMAN BACKUP command supports backing up the following types of files:

- Data files and control files
- Server parameter file
- Archived redo logs
- RMAN backups

RMAN doesn't backup everything associated with the Oracle database.

Although the database depends on other types of files, such as network configuration files, password files, and the contents of the Oracle home, you cannot back up these files with RMAN. Likewise, some features of Oracle Database, such as external tables, may depend upon files other than the data files, control files, and redo log. The files listed below are a minimum that of files to be copied to PowerVS outside of RMAN backup procedures:

- Oracle database parameter file: init<SID>.ora
- TNS network configuration files: (eg. listener.ora, tnsnames.ora)
- Oracle database password file, if used.

RMAN does not backup these files.

When you execute the BACKUP command in RMAN, the output is always either one or more backup sets or one or more image copies. A [backup set](#) is an RMAN-specific proprietary format, whereas an [image copy](#) is a bit-for-bit copy of a file. By default, RMAN creates backup sets. This document only deals with backup sets.

Assumptions

This document assumes that

- If the source database is a production system it likely won't be shut down for a full backup, in which case RMAN incremental backups will be used.
- No changes to the source database (schema) are expected by executing this procedure.
- Database will be migrated from source to target using Oracle RMAN options:
 - Duplicate database, or
 - backup/restore/recover database.
- Target environment will have Oracle homes pre-installed with a version matching the source instance being migrated.
- RMAN duplicate, restore/recover provide the option to modify the database data file location, but this option is not covered in this document. The assumption is made that the location of database files, either AIX JFS2 file systems or Oracle ASM disk groups, is identical between the source and the target environment.
- Target OS will be either
 - an exact copy (an OVA or "Open Virtual Appliance" archive file) of the source operating system generated using a **mksysb** procedure, or

- a freshly deployed version of AIX that is supported by IBM and certified by Oracle to support the version of Oracle to be instantiated.

Documentation of a **mysysb** procedure may be found at <https://cloud.ibm.com/docs/power-iaas?topic=power-iaas-restoring-aix-mksysb-image>

Preparation

Discovery has confirmed or identified the following:

1. Configuration of the source Oracle instance(s), and has documented current sizing and performance metrics related to compute, IOPS and storage.
 - The target PowerVS Oracle instance should be constructed to at least match the sizing of the existing system.
 - Very important on the target system to select disk tier and disk capacity to meet IOPS requirements
 - Discovery process should include execution of a performance test tool against the source database to capture metrics for future reference for target system deployment comparison.
2. Before work begins, Customer should confirm, using their standard backup tools, that a recent, successful, full backup of the source Oracle system exists, and that, if necessary, Customer could completely restore the source system in the event of a software issue.
 - If unable to confirm, Discovery team should review RMAN backup procedures with the Customer.
 - In addition, Discovery team should work with Customer DBA or SME to validate database integrity and/or identify any existing corrupt blocks or schema problems.
3. Customer has provided location of source database instance(s) and credentials for administrative access.
4. Discovery team has identified and documented the current configuration of RMAN that is used by the Customer to perform backups, per instance.
 - This can be obtained by performing the following:
 - Use ssh to connect to the Oracle Database instance.
 - You can connect to a database with command-line options. or by using the CONNECT TARGET command.
 - Use the following command to connect to the source database with RMAN. (This command assumes that the ORACLE_SID is set.)

```
# rman target /  
RMAN> SHOW ALL;
```
 - Store and/or document output
 - Enter EXIT to exit the RMAN session.
5. Confirm enough disk space is available, formatted, and mounted to accept compressed “migration” database backup sets.
 - Best performance is obtained if the disk space is local to the instance being backed up.

- This procedure will perform a backup, applying medium compression with the following considerations:
 - Compression will result in backup file storage that takes approximately 25% of the space consumed by the Oracle database; but note that this is dependent on database content and needs to be verified in customer environment.
 - Local file system capacity should be sufficient to store the entire backup set (or sets) required to bring target database to the restore point desired.
 - A local file system for Power access (/backup/rman) for the migration backup sets is referenced below.
 - Procedures were also successfully tested using direct restore/recovery from an NFS mounted Seagate Lyve Mobile device using 10GbE and 100GbE.

- The duplicate or restore/recover procedures will require access to the set of backup files, via one of the following options:
 - A local filesystem to which the backup sets will be copied,
 - An NFS mount to a Seagate Lyve Mobile transfer device, or
 - An NFS mount to IBM Cloud Object Storage to which the backup files have already been copied from the Seagate device.
 - Note: Depending on network/access speed, it may be desirable to copy files from NFS mount to local storage.

RMAN Backups

Reference:

- <https://docs.oracle.com/en/database/oracle/oracle-database/19/rcmf/database-backup-and-recovery-reference.pdf> (September 2023)

Assumptions

Option 1 procedure assumes that a consistent offline backup will be taken of the source database.

Option 2 procedure requires that the source database is in ARCHIVELOG mode before the backup procedure is performed.

Considerations

Customer should balance how parallelism and/or compression is applied (and associated resources allocated to support) with requirements for database availability and performance while backups are being performed.

Consider applying “section size” to backup configuration. Without specifying “section size” you can end up with very few very large files which makes it challenging to handle or restart in case of a transfer failure. Parallelism at restore is also limited to the number of backup files available to process concurrently. Using a “good” section size allows to control the size of each backup file generated and thereby also influence how many files are generated which can then be processed in parallel during restore in PowerVS.

Note that *Medium* compression requires Oracle Advanced Compression License. *Basic* compression is “free”, but significantly slower and achieves lower compression rates. *High* compression, is VERY CPU intensive on the compression side and only provides somewhat limited compression benefit as compared to *Medium*. *High* compression also requires the Advanced Compression License.

Incremental backup, without database block change tracking activated, performs a full scan of all data files, which translates into a very high read workload. Database block change tracking can significantly reduce that read workload. Although enabling block change tracking is not expected to impact performance of a running DB, testing should be performed in the Customer environment to validate this.

Network Transfer – Standard Protocols vs. Aspera

You can use standard protocols (such as sftp) to transmit the backup files either directly to PowerVS accessible storage, or to Cloud Object Storage. Using sftp assumes that you have installed and configured a sftp server within or adjacent to the target PowerVS environment to receive the transfer.

If available, you can use IBM’s high performance Aspera product to more quickly effect the transfers. In many situations, Aspera has been shown to transfer data several times faster than traditional TCP-based protocols.

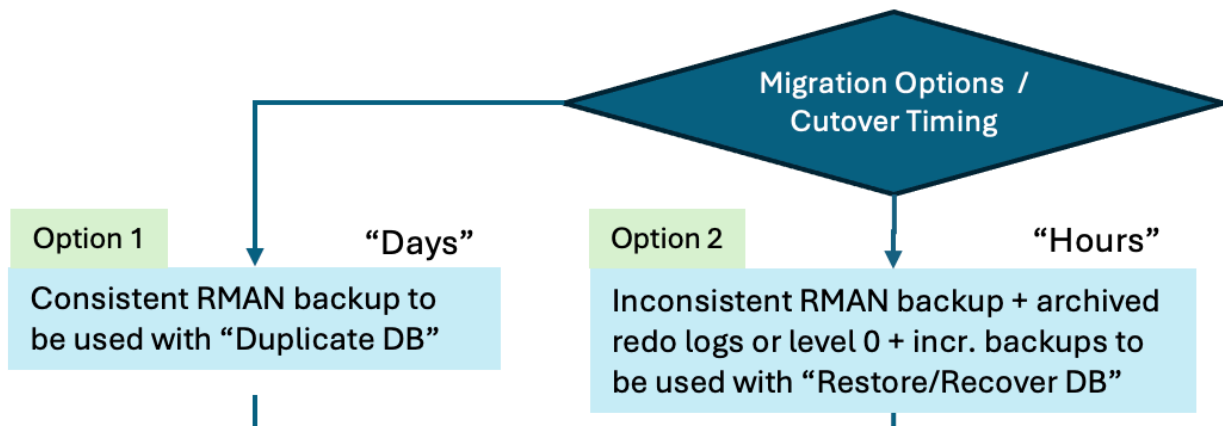
Documentation for Aspera use can be found here:

<https://cloud.ibm.com/docs/power-iaas?topic=power-iaas-migration-strategies-power#aspera-technologies>

- Reference the “migration strategy” page under Aspera technologies.

Procedure

Backup procedure varies depending on when you are using Option1 or Option 2.



Procedures specific to each option are presented below.

Option 1: RMAN Duplicate Database

The following procedure is executed with the database offline and will produce a full consistent backup.

1. Use standard operational procedures to shutdown the database before performing the backup procedure.
2. When using the multitenant architecture, you must connect to the root CDB and the backups will include the pluggable databases (PDBs). More information can be found in [About Performing Operations on CDBs and PDBs](#).
3. **Be sure to record current RMAN configuration and associated parameters.** After completing special backups for migration, you want to ensure RMAN configuration is left as before so normally scheduled backup operations continue.
 - o Connect to RMAN and perform **SHOW ALL** to see existing configuration. Parameters will be added/adjusted to perform the backup for the database migration in the next step.

```

# export ORACLE_SID=ECOM
# rman target /

Recovery Manager: Release 10.0.0.0.0 - Production on Tue Nov 14 09:01:24 2023
Version 10.21.0.0.0

Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.

Connected to target database: ECOM (DBID=3462477967)

RMAN> show all ;
Using target database control file instead of recovery catalog
RMAN configuration parameters for database with db_unique_name ECOM are:
CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default
CONFIGURE BACKUP OPTIMIZATION OFF; # default
CONFIGURE DEFAULT DEVICE TYPE TO DISK; # default
CONFIGURE CONTROLFILE AUTOBACKUP ON; # default
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO '%F'; # default
CONFIGURE DEVICE TYPE DISK PARALLELISM 1 BACKUP TYPE TO BACKUPSET; # default
CONFIGURE DATAFILE BACKUP COPIES FOR DEVICE TYPE DISK TO 1; # default
CONFIGURE ARCHIVELOG BACKUP COPIES FOR DEVICE TYPE DISK TO 1; # default
CONFIGURE MAXSETSIZE TO UNLIMITED; # default
CONFIGURE ENCRYPTION FOR DATABASE OFF; # default
CONFIGURE ENCRYPTION ALGORITHM 'AES128'; # default
CONFIGURE COMPRESSION ALGORITHM 'BASIC' AS OF RELEASE 'DEFAULT' OPTIMIZE FOR LOAD TRUE; # default
CONFIGURE RMAN OUTPUT TO KEEP FOR 7 DAYS; # default
CONFIGURE ARCHIVELOG DELETION POLICY TO NONE; # default
CONFIGURE SNAPSHOT CONTROLFILE NAME TO '/ora/db/19c/dbs/snapcf_ecom.f'; # default

```

Exit RMAN after collecting the information.

4. Execute the following as oracle user to create the target backup directory to match the following backup script. If the directory does not exist the rman script will fail. The second command will set <ORACLE_SID>.

```

# mkdir -p /backup/rman/ecom_option1
# export ORACLE_SID=ECOM

```

5. The RMAN script shown below is used to execute the backup. The content of the script for this Option 1 is discussed further down in this document.

option1_backup.rman

```

connect target /
SHUTDOWN IMMEDIATE
STARTUP MOUNT
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT '/backup/rman/ecom_option1/option1_%d_%U';
CONFIGURE DEVICE TYPE DISK PARALLELISM 60;
CONFIGURE CONTROLFILE AUTOBACKUP ON;
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT
  FOR DEVICE TYPE DISK TO '/backup/rman/ecom_option1/option1_cf_%F';
CONFIGURE COMPRESSION ALGORITHM 'MEDIUM';
CONFIGURE ENCRYPTION FOR DATABASE ON ;
CONFIGURE ENCRYPTION ALGORITHM 'AES256' ;
SET ENCRYPTION ON IDENTIFIED BY passw0rd ONLY ;
BACKUP AS compressed BACKUPSET section size 6000M DEVICE
  TYPE DISK DATABASE PLUS ARCHIVELOG
  TAG ECOM_option1 include current controlfile;
quit;

```

You can call the script directly from the command line using

```
# rman @option1_backup.rman
```

RMAN backs up data to the configured default device for the type of backup requested. By default, RMAN creates backups on disk. If a fast recovery area is enabled, and if you do not specify the FORMAT parameter, then RMAN creates backups in the recovery area and automatically gives them unique names. This is the reason for modifying FORMAT in the parameters above and is repeated here for emphasis.

The following are the main components of the rman script used:

- For Option 1 we need to have the database cleanly shut down and then started in “mount mode” and not open to ensure that the backup is consistent and can be restored.
 - SHUTDOWN IMMEDIATE
 - STARTUP MOUNT
- The restore of the database requires a copy of the database control file as it contains the RMAN catalog required to restore the backup pieces into a functioning database.
 - CONFIGURE CONTROLFILE AUTOBACKUP ON ;
 - BACKUP AS compressed BACKUPSET section size 6000M DEVICE TYPE DISK DATABASE PLUS ARCHIVELOG TAG ECOM_option1 include current controlfile;
- Note that the CONFIGURE CHANNEL DEVICE TYPE DISK and CONFIGURE CONTROLFILE AUTOBACKUP FORMAT for DEVICE TYPE DISK commands in the above script include the backup file system location.
- Set disk device parallelism – likely to help reduce backup window. The optimal parallelism is dependent on several factors:
 - Availability of CPU resources to run that many concurrent backup processes. With the selected compression and encryption each RMAN process typically uses all CPU cycles of a logical processor, assuming the storage subsystem can provide the data fast enough.
 - Capability of storage subsystem to support the RMAN data file read and the write to backup location I/O throughput.
 - Amount of free physical memory to support the backup processes to read, compress, encrypt the data.
 - Size of the database.
- Parallelism 60 was used in the our testing as shown in the rman script above, but a parallelism of 8, as shown below, may be good starting point to find an optimal level during discovery.
 - CONFIGURE DEVICE TYPE DISK PARALLELISM 8 BACKUP TYPE TO BACKUPSET;

- Set backup file compression. MEDIUM is shown, HIGH could be an option under certain circumstances. The use of 'MEDIUM' and 'HIGH' requires the Oracle Advanced Compression license!
 - CONFIGURE COMPRESSION ALGORITHM 'MEDIUM' ;
- Set backup encryption on and modify algorithm if desired (default is AES128) and specify the encryption password to be used. Unless you are working with a TDE-encrypted database, every RMAN session requires the setting of the encryption and decryption password; otherwise, the session fails with a "wallet not open" error.
 - CONFIGURE ENCRYPTION FOR DATABASE ON ;
 - CONFIGURE ENCRYPTION ALGORITHM 'AES256' ;
 - SET ENCRYPTION ON IDENTIFIED BY passw0rd ONLY ;
- The final command then triggers the actual backup of the database to disk. The backup uses the RMAN specific backup sets. Each RMAN process will read up to 6000M of consecutive data from a data file as a backup piece, compress and encrypt that data and then write it to the destination file system.

The goal of the section size is to limit the size of a backup piece so that a potentially required retransmit of a failed file transfer to the destination environment is manageable. The use of a TAG is strongly recommended as it simplifies the management of multiple backups in a RMAN catalog.

- BACKUP AS compressed BACKUPSET section size 6000M DEVICE TYPE DISK DATABASE PLUS ARCHIVELOG TAG ECOM_option1 include current controlfile;

6. When backup is complete, within RMAN, you can use some of the commands listed below to validate and cross check results.

```
RMAN> REPORT SCHEMA;
RMAN> LIST BACKUP SUMMARY;
RMAN> BACKUP VALIDATE CHECK LOGICAL DATABASE ARCHIVELOG ALL;
RMAN> VALIDATE DATAFILE 10;
RMAN> VALIDATE BACKUPSET 3;
RMAN> CROSSCHECK BACKUP;
```

Explanation of commands:

- REPORT SCHEMA: Lists and displays information about the database files, tablespaces, etc.
- LIST BACKUP: Lists all existing backups. A SUMMARY option can be used.
- BACKUP VALIDATE CHECK LOGICAL DATABASE ARCHIVELOG ALL: Validate the contents of backup files.
- VALIDATE DATAFILE 10: Validate a specific datafile.
- VALIDATE BACKUPSET 3: Validate a specific backupset.
- CROSSCHECK BACKUP: Synchronize the physical reality of backups and copies with their logical records in the RMAN repository.

7. Create a plain text database parameter file (PFILE) from the SPFILE as follows while the database is in mounted mode.

```
# export ORACLE_SID=ECOM
# sqlplus / as sysdba;
SQL> create pfile='/backup/ora/initECOM.ora' from spfile;
```

Be sure to restore all RMAN parameters back to original discovery settings when done.

For this Option 1 we have created ONE backup, to be transferred to and restored on the target system. At this point, to be consistent with a migration scenario, the source database should be shut down and not used any longer.

Option 2: RMAN Restore/Recover Database

The following backup procedure is executed with the database online, and will produce a single RMAN Level 0 and one or more Level 1 (differential) backups. It is required that the databases be in archive log mode and that it is ensured that all required archived redo logs are included in the backups.

- An RMAN Incremental Level 0 backup is a full backup.
 - An RMAN Incremental Level 1 backup is an incremental backup, capturing the changes since the previous Level 0 or Level 1 backup.
1. Note that the level 0 and incremental backups are taken with the database online, except potentially the last incremental, and that archived redo logs are required to restore and recover the database successfully.
 2. When using the multitenant architecture, you must connect to the root CDB and the backups will include the pluggable databases (PDBs). More information can be found in [About Performing Operations on CDBs and PDBs](#).
 3. **Be sure to record current RMAN configuration and associated parameters.** After completing special backups for migration, you want to ensure RMAN configuration is left as before so normally scheduled backup operations continue.
 - Connect to RMAN and perform **SHOW ALL** to see existing configuration. Parameters will be added/adjusted to perform the backup for the database migration in the next step.

```
# export ORACLE_SID=ECOM
# rman target /

Recovery Manager: Release 10.0.0.0.0 - Production on Tue Nov 14 09:01:24 2023
Version 10.21.0.0.0

Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.

Connected to target database: ECOM (DBID=3462477967)

RMAN> show all ;
Using target database control file instead of recovery catalog
RMAN configuration parameters for database with db_unique_name ECOM are:
CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default
CONFIGURE BACKUP OPTIMIZATION OFF; # default
CONFIGURE DEFAULT DEVICE TYPE TO DISK; # default
CONFIGURE CONTROLFILE AUTOBACKUP ON; # default
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO '%F'; # default
CONFIGURE DEVICE TYPE DISK PARALLELISM 1 BACKUP TYPE TO BACKUPSET; # default
CONFIGURE DATAFILE BACKUP COPIES FOR DEVICE TYPE DISK TO 1; # default
CONFIGURE ARCHIVELOG BACKUP COPIES FOR DEVICE TYPE DISK TO 1; # default
CONFIGURE MAXSETSIZE TO UNLIMITED; # default
CONFIGURE ENCRYPTION FOR DATABASE OFF; # default
CONFIGURE ENCRYPTION ALGORITHM 'AES128'; # default
CONFIGURE COMPRESSION ALGORITHM 'BASIC' AS OF RELEASE 'DEFAULT' OPTIMIZE FOR LOAD TRUE; #
default
CONFIGURE RMAN OUTPUT TO KEEP FOR 7 DAYS; # default
CONFIGURE ARCHIVELOG DELETION POLICY TO NONE; # default
CONFIGURE SNAPSHOT CONTROLFILE NAME TO '/ora/db/19c/dbs/snapcf_ecom.f'; # default
```


Exit RMAN after collecting the information.

4. Execute the following as oracle user to create the target backup directory to match the following backup script and set <ORACLE_SID>.

```
# mkdir -p /backup/rman/ecom_option2
# export ORACLE_SID=ECOM
```

5. The RMAN script shown below was used to execute the initial level 0 (full) online backup. The content of the script is discussed further down in this document.

option2_backup_level0.rman

```
connect target /
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT '/backup/rman/ecom_option2/lev0_%d_%U';
CONFIGURE BACKUP OPTIMIZATION ON;
CONFIGURE DEVICE TYPE DISK PARALLELISM 60;
CONFIGURE CONTROLFILE AUTOBACKUP ON;
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT
  FOR DEVICE TYPE DISK TO '/backup/rman/ecom_option2/lev0_cf_%F';
CONFIGURE COMPRESSION ALGORITHM 'MEDIUM';
CONFIGURE ENCRYPTION FOR DATABASE ON ;
CONFIGURE ENCRYPTION ALGORITHM 'AES256' ;
SET ENCRYPTION ON IDENTIFIED BY passw0rd ONLY ;
BACKUP incremental level 0 AS compressed BACKUPSET section size 6000M DEVICE TYPE DISK DATABASE
  INCLUDE CURRENT CONTROLFILE PLUS ARCHIVELOG TAG ECOM_LEV0;
quit;
```

You can call the script directly from the command line using

```
# rman @option2_backup_level0.rman
```

RMAN backs up data to the configured default device for the type of backup requested. By default, RMAN creates backups on disk. If a fast recovery area is enabled, and if you do not specify the FORMAT parameter, then RMAN creates backups in the recovery area and automatically gives them unique names. This is the reason for modifying FORMAT in the parameters above and is repeated here for emphasis.

The following are the main components of the rman script used:

- o RMAN, by default, includes ALL archived redo logs in a level 0 or level 1 backup if the “PLUS ARCHIVELOG” flag is specified. This can result in many duplicates of the same file within a consecutive set of incremental backups.

The following directive instructs RMAN to check if a specific archived redo log was already included in a previous backup and, if yes, do not include the file in a new incremental backup. Note that this option should NOT be used if other backups are taken of the database which are not to be transferred to the destination.

- CONFIGURE BACKUP OPTIMIZATION ON;

- The restore of the database requires a copy of the database control file as it contains the RMAN catalog required to restore the backup pieces into a functioning database.
 - `CONFIGURE CONTROLFILE AUTOBACKUP ON;`
 - `BACKUP AS compressed BACKUPSET section size 6000M DEVICE TYPE DISK DATABASE PLUS ARCHIVELOG TAG ECOM_option1 include current controlfile;`
- Note that the `CONFIGURE CHANNEL DEVICE TYPE DISK` and `CONFIGURE CONTROLFILE AUTOBACKUP FORMAT` for `DEVICE TYPE DISK` commands in the above script include the backup file system location.
- Set disk device parallelism – likely to help reduce backup window. The optimal parallelism is dependent on several factors:
 - Availability of CPU resources to run many concurrent backup processes. With the selected compression and encryption each RMAN process typically uses all CPU cycles of a logical processor, assuming the storage subsystem can provide the data fast enough.
 - Capability of storage subsystem to support the RMAN data file read and write to backup location I/O throughput.
 - Amount of free physical memory to support the backup processes to read, compress, encrypt the data.
 - Size of the database.
- Parallelism 60 was used in the our testing as shown in the rman script above, but a parallelism of 8, as shown below, may be good starting point to find an optimal level during discovery.
 - `CONFIGURE DEVICE TYPE DISK PARALLELISM 8 BACKUP TYPE TO BACKUPSET;`
- Set backup file compression. `MEDIUM` is shown, `HIGH` could be an option under certain circumstances. The use of '`MEDIUM`' and '`HIGH`' requires the Oracle Advanced Compression license!
 - `CONFIGURE COMPRESSION ALGORITHM 'MEDIUM' ;`
- Set backup encryption on and modify algorithm if desired (default is `AES128`) and specify the encryption password to be used. Unless you are working with a TDE-encrypted database, every RMAN session requires the setting of the encryption and decryption password; otherwise, the session fails with a "wallet not open" error.
 - `CONFIGURE ENCRYPTION FOR DATABASE ON;`
 - `CONFIGURE ENCRYPTION ALGORITHM 'AES256';`
 - `SET ENCRYPTION ON IDENTIFIED BY passw0rd ONLY ;`
- The final command then triggers the actual backup of the database to disk. The backup uses the RMAN specific backup sets. Each RMAN process will read 6000M of consecutive data from a data file as a backup piece, compress and encrypt it and then write it to the destination file system. The goal of the section size is to limit the size of a backup piece so that a potentially required retransmit of a failed file transfer to the destination environment is manageable.

As pointed out before, it is essential that archived redo log files are included in the backup and the `PLUS ARCHIVELOG` flag ensures that. The use of a TAG is strongly recommended as it simplifies the management of multiple backups in a RMAN catalog. “Incremental level 0” specifies this backup as an incremental backup at level 0, means all data will be included in the backup.

- `BACKUP incremental level 0 AS compressed BACKUPSET section size 6000M DEVICE TYPE DISK DATABASE INCLUDE CURRENT CONTROLFILE PLUS ARCHIVELOG TAG ECOM_LEV0;`

For additional incremental backups we suggest to replace occurrences of “inc1 / INC1” with corresponding “incN / INCN” in the following backup script. Note that it is suggested to store each incremental set of backup files into its own directory for easier management.

5. Execute the following as oracle user to create the target backup directory to match the backup script for the first incremental level 1 backup and set `<ORACLE_SID>`.

```
# mkdir -p /backup/rman/ecom_option2_inc1
# export ORACLE_SID=ECOM
```

option2_backup_inc1.rman

```
connect target /
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT '/backup/rman/ecom_option2_inc1/inc1_%d_%U';
CONFIGURE BACKUP OPTIMIZATION ON;
CONFIGURE DEVICE TYPE DISK PARALLELISM 60;
CONFIGURE CONTROLFILE AUTOBACKUP ON;
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT
  FOR DEVICE TYPE DISK TO '/backup/rman/ecom_option2_inc1/inc1_cf_%F';
CONFIGURE COMPRESSION ALGORITHM 'MEDIUM';
CONFIGURE ENCRYPTION FOR DATABASE ON ;
CONFIGURE ENCRYPTION ALGORITHM 'AES256' ;
SET ENCRYPTION ON IDENTIFIED BY passwd ONLY ;
BACKUP incremental level 1 AS compressed BACKUPSET section size 6000M DEVICE TYPE DISK DATABASE
  INCLUDE CURRENT CONTROLFILE PLUS ARCHIVELOG TAG ECOM_INC1;
quit;
```

You can call the script directly from the command line using

```
# rman @option2_backup_inc1.rman
```

The incremental level 1 scripts differ from the previously discussed level 0 rman script only in 3 areas:

- i. The directory where the backup files are written to.
- ii. The TAG used to identify the backup set

- iii. The specification of `level 1` instead of `level 0`, indicating that this is an incremental backup, which by default, is a differential backup to the last backup taken.

The final incremental backup requires additional preparation steps to determine the date/time stamp to use in the final database recovery step for the destination environment. The time stamp highlighted in **blue box** in the next step is the date/time the database will be recovered to. Any changes after that date/time will be discarded!

6. Connect to the database as `sysdba` and execute these commands, as shown below. Note that the database is only put into “mount” mode and not “opened”.

```
# sqlplus / as sysdba
SQL> alter session set nls_date_format='MM-DD-YYYY HH24:mi:ss' ;
      Session altered.

SQL> SELECT SYSDATE FROM DUAL ;
      SYSDATE
      -----
      01-13-2024 11:51:24

SQL> ALTER SYSTEM SWITCH LOGFILE ;
      System altered.

SQL> SELECT SYSDATE FROM DUAL ;
      SYSDATE
      -----
      01-13-2024 11:52:12

SQL> ALTER SYSTEM SWITCH LOGFILE ;
      System altered.

SQL> SELECT SYSDATE FROM DUAL ;
      SYSDATE
      -----
      01-13-2024 11:53:14
SQL> shutdown immediate
SQL> startup mount
```

Execute the final incremental backup via RMAN.

7. When backup is complete, within RMAN, you can utilize some of the commands listed below to validate and cross check results.

```
RMAN> REPORT SCHEMA;
RMAN> LIST BACKUP;
RMAN> BACKUP VALIDATE CHECK LOGICAL DATABASE ARCHIVELOG ALL;
RMAN> VALIDATE DATAFILE 10;
RMAN> VALIDATE BACKUPSET 3;
RMAN> CROSSCHECK BACKUP;
```

Explanation of commands:

- `REPORT SCHEMA`: Lists and displays information about the database files, tablespaces, etc.
- `LIST BACKUP`: Lists all existing backups. A `SUMMARY` option can be used.

- BACKUP VALIDATE CHECK LOGICAL DATABASE ARCHIVELOG ALL: Validate the contents of backup files.
- VALIDATE DATAFILE 10: Validate a specific datafile.
- VALIDATE BACKUPSET 3: Validate a specific backupset.
- CROSSCHECK BACKUP: Synchronize the physical reality of backups and copies with their logical records in the RMAN repository.

8. Create a plain text database parameter file (PFILE) from the SPFILE as follows while the database is in mounted mode.

```
# export ORACLE_SID=ECOM
# sqlplus / as sysdba;
SQL> create pfile='/backup/ora/initECOM.ora' from spfile;
```

Be sure to restore all RMAN parameters back to original discovery settings when done.

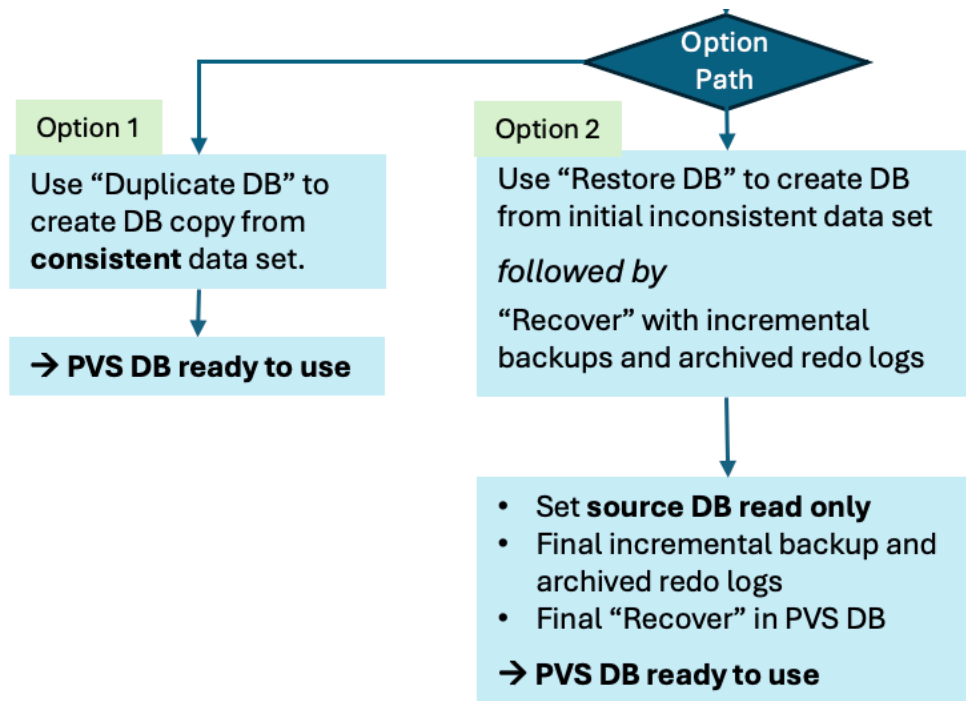
RMAN Restore

Two Restore Options

Two methods to restore an Oracle Database to the target system are provided:

- 1) RMAN Duplicate Database and
- 2) RMAN Restore/Recover Database.

Both methods will use the backups created in the procedure described above.



Option 1: RMAN Duplicate Database

This option will use the “auxiliary instance” function of RMAN and can/should be used only when the full set of the consistent RMAN backup files are available on the target server. The RMAN “duplicate” command will automatically open the target database with “resetlogs”. The resetlogs operation increments the database incarnation number and, as a result, makes the target database completely independent of the source. Importantly, no incremental level 1 backups or archived REDO logs can be applied to the target database at that point.

Option 2: RMAN Restore/Recover Database

This option should be used to initially establish the target database with an RMAN incremental level 0 and, optionally, one or more incremental level 1 backup(s) as described in this document. However, it allows for later incremental level 1 backups and archived REDO logs to be applied to the target database over some amount of time.

The expectation is that there will be a final cutover time, and the final incremental level 1 backup will be taken, copied/shipped to the target server and applied while the target database is still in a MOUNT state. Once the final backup/archived logs are applied, the database will be manually opened and made available at the cutover time.

Assumptions

- An IBM cloud account has been established, along with a PowerVS workspace,
- Within the workspace, manual or automated procedures have been executed to install on the PowerVS instance the necessary Oracle Grid Infrastructure (if using ASM) and Oracle RDBMS homes on AIX. The AIX LPAR should be sized comparably to the source system in terms of compute, memory, storage capacity and performance, with compatible HW/SW versioning applied.
- All required backup files are accessible to the RMAN duplicate/restore/recover process.

Connectivity and Preparation – Oracle on PowerVS

From the target instance command line, you can use the following RMAN procedure to perform a restore.

1. Login as the oracle user to the AIX Power Virtual Server where you want to restore the database.
2. Confirm that
 - Target PowerVS system has a valid install of the Oracle database software with the same version and patch level of software as in the source environment.
 - The device names for storage of Oracle datafiles and tablespace names are the same as with the source database. If not, “set newname” directives will be required (steps are out of scope for this procedures)
 - ASM disk groups (if used) have been established with names and capacities that match the source environment.
 - Backup files are available to RMAN, whether resident on local file storage, Cloud Object Storage, NFS, etc.

The following procedure options do not describe how to install the Oracle software, but rather how to restore the source database. As with the backup procedure, the presented steps are representative, and details will differ from one database environment to another.

Please reference the version-specific Oracle Database Installation Guide for AIX on Power Systems and the corresponding GRID Infrastructure documentation for ASM (and RAC, if relevant).

Database Installation Guide (Version 19c):

<https://docs.oracle.com/en/database/oracle/oracle-database/19/axdbi/index.html#Oracle%C2%AE-Database>

Grid Infrastructure Documentation (Version 19c):

<https://docs.oracle.com/en/database/oracle/oracle-database/19/cwaix/installing-oracle-grid-infrastructure.html#GUID-D4E3FADF-360E-49EB-89A2-E4CBBB9CC61F>

Procedure

Here is a sample view of backup files that have been staged in the /backup/rman folder:
Note that files were copied to local JFS/2 file system... partial list of files.


```

ls -ltr /backup/rman/ecom_option1
total 498717736
-rw-r--r-- 1 oracle oinstall 1089536 Dec 06 18:20 option1_n92dccnp_745_1_1
-rw-r--r-- 1 oracle oinstall 1081344 Dec 06 18:20 option1_n82dccnp_744_1_1
-rw-r--r-- 1 oracle oinstall 1089536 Dec 06 18:20 option1_n72dccnp_743_1_1
-rw-r--r-- 1 oracle oinstall 1146880 Dec 06 18:20 option1_n62dccnp_742_1_1
-rw-r--r-- 1 oracle oinstall 4816896 Dec 06 18:20 option1_mt2dccno_733_1_1
-rw-r--r-- 1 oracle oinstall 1187840 Dec 06 18:20 option1_ms2dccno_732_1_1
-rw-r--r-- 1 oracle oinstall 1187840 Dec 06 18:20 option1_mr2dccno_731_1_1
-rw-r--r-- 1 oracle oinstall 1073152 Dec 06 18:20 option1_ne2dccnq_750_1_1
...
-rw-r--r-- 1 oracle oinstall 1081327616 Dec 06 18:42 option1_mq2dccno_730_71_1
-rw-r--r-- 1 oracle oinstall 1084383232 Dec 06 18:42 option1_mq2dccno_730_69_1
-rw-r--r-- 1 oracle oinstall 1084596224 Dec 06 18:42 option1_mq2dccno_730_64_1
-rw-r--r-- 1 oracle oinstall 1084186624 Dec 06 18:42 option1_mq2dccno_730_73_1
-rw-r--r-- 1 oracle oinstall 1084547072 Dec 06 18:42 option1_mq2dccno_730_74_1
-rw-r--r-- 1 oracle oinstall 1084596224 Dec 06 18:42 option1_mq2dccno_730_78_1
-rw-r--r-- 1 oracle oinstall 1084588032 Dec 06 18:42 option1_mq2dccno_730_79_1
-rw-r--r-- 1 oracle oinstall 1084743680 Dec 06 18:42 option1_mq2dccno_730_76_1
-rw-r--r-- 1 oracle oinstall 1084530688 Dec 06 18:42 option1_mq2dccno_730_75_1
-rw-r--r-- 1 oracle oinstall 1072553984 Dec 06 18:42 option1_mq2dccno_730_77_1
-rw-r--r-- 1 oracle oinstall 14614528 Dec 06 18:42 option1_cf_c-3935091107-20231206-02

```

Displayed below in a separate folder are miscellaneous files directly copied from the on-premises Oracle database, as recommended in the backup procedure above.

```

# cd /backup
# ls -l
total 32
-rw-r--r-- 1 65534 65334 6956 Dec 06 07:56 checksums # optionally saved cksum information for backup files
drwxrwxrwx 2 65534 65534 4096 Dec 06 07:08 ora # saved parameter file (PFILE) and TNS configuration
drwxrwxrwx 2 65534 65534 4096 Dec 06 07:08 rman # contains the set of RMAN incremental backup files

(0) Root @ isstecom01: /backup/ora
# ls
initECOM.ora listener.ora. sqlnet.ora tnsnames.ora

```

We will now review the Oracle parameter file, or “PFILE”, stored within as “initECOM.ora”.

1. Review database parameters contained within the saved PFILE and examine those with file name specifications. The parameter “audit_file_dest=” references a directory location that does not yet exist on the target server. The other file destinations reference ASM disk groups that should have already been created. If the file locations in the target server do not match the source environment additional configuration changes are required in the rman scripts, but those changes are not covered here.

```

*.ash_size=536870912
*.audit_file_dest='/u01/app/oracle/admin/ecom/adump'
*.audit_trail='db'
*.compatible='19.0.0'
*.control_files='+REDO1DG/ECOM/CONTROLFILE/current.258.1158057959','+REDO2DG/ECOM/CONTROLFILE/cu
rrent.258.1158057959'#Restore Controlfile
*.db_block_size=8192
*.db_cache_size=30G
*.db_create_file_dest='+DATA'
*.db_create_online_log_dest_1='+REDO1DG'
*.db_create_online_log_dest_2='+REDO2DG'
*.db_domain='pvs_domain'
*.db_name='ecom'
*.db_recovery_file_dest='+ARCH'
*.db_recovery_file_dest_size=350g
*.diagnostic_dest='/u01/app/oracle'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=ECOMXDB) '
*.enable_pluggable_database=true
*.filesystemio_options='SETALL'
*.local_listener='LISTENER_ECOM'
*.nls_language='AMERICAN'
*.nls_territory='AMERICA'
*.open_cursors=600
*.parallel_max_servers=128
*.parallel_min_servers=16
*.parallel_servers_target=16
*.pga_aggregate_target=5G
*.processes=16384
*.remote_login_passwordfile='EXCLUSIVE'
*.sga_max_size=55G
*.sga_target=40G
*.undo_tablespace='UNDOTBS1'

```

2. Create the audit file directory as listed in the PFILE and give the proper ownership and mode.

```

# grep audit_file initECOM.ora
*.audit_file_dest='/u01/app/oracle/admin/ecom/adump'

# mkdir -p /u01/app/oracle/admin/ecom/adump
# chown -R oracle:oinstall /u01/app/oracle/admin/ecom/adump

```

3. Confirm that ASM disk groups referenced in the PFILE exist on target and have sufficient free space.

As grid user, run: **asmcmd lsdg**

```
# asmcmd lsdg
State Type   Rebal Sector Logical_Sector Block      AU   Total_MB   Free_MB   Req_mir_free_MB   Usable_file_MB   Offline_disks
Voting_files Name
MOUNTED EXTERN N      4096      4096  4096 4194304    143040    142920      0      142920      0
N   ARCH/
MOUNTED EXTERN N      4096      4096  4096 4194304    2956700    2956548      0      2956548      0
N   DATA/
MOUNTED EXTERN N      4096      4096  4096 4194304     95360     95252      0      95252      0
N   REDO1DG/
MOUNTED EXTERN N      4096      4096  4096 4194304     95360     95252      0      95252      0
N   REDO2DG/
```

If ASM groups are not used, review the PFILE and verify that referenced directories exist.

At this point, you are ready to proceed with one of the following restore options.

Option 1: RMAN Duplicate Database

This procedure takes as input a consistent, Level 0 backup of the source database and restores the contents to a new Oracle instance on Power Virtual Server to create a duplicate database.

The following steps should be executed as the oracle user:

1. Set appropriate ORACLE_SID name.

```
# export ORACLE_SID=ECOM
```

2. Start the Oracle instance in NOMOUNT mode, directly specifying the pfile to be used. Note that in our environment ORACLE_HOME was “/u01/app/oracle”, your environment may differ.

```
connect / as sysdba
startup nomount pfile='/backup/ora/initECOM.ora'
create spfile='/u01/app/oracle/dbs/spfileECOM.ora' from pfile='/backup/ora/initECOM.ora';
shutdown immediate
startup nomount
show parameter spfile
```

NAME	TYPE	VALUE
spfile	string	/u01/app/oracle/dbs/spfileECOM.ora

3. Create RMAN Duplicate Database script referencing the appropriate backup location. The file in this case is stored in the home directory of the ‘oracle’ user.

Optimally, you would create the number of restore channels below to match the number of channels used for backup. In the example below, we assume 15 channels were used for backup.

rman_duplicate.cmd

```
set encryption on identified by passw0rd;
set decryption identified by passw0rd;
run {
allocate auxiliary channel ch1 device type disk;
allocate auxiliary channel ch2 device type disk;
allocate auxiliary channel ch3 device type disk;
allocate auxiliary channel ch4 device type disk;
allocate auxiliary channel ch5 device type disk;
allocate auxiliary channel ch6 device type disk;
allocate auxiliary channel ch7 device type disk;
allocate auxiliary channel ch8 device type disk;
allocate auxiliary channel ch9 device type disk;
allocate auxiliary channel ch10 device type disk;
allocate auxiliary channel ch11 device type disk;
allocate auxiliary channel ch12 device type disk;
allocate auxiliary channel ch13 device type disk;
allocate auxiliary channel ch14 device type disk;
allocate auxiliary channel ch15 device type disk;
duplicate database to ECOM backup location '/backup/rman/ ecom_option1';
}
```

```
# rman auxiliary / cmdfile=rman_duplicate.cmd
```

Sample output:

```
oracle@isstecom01:/home/oracle -> rman auxiliary / cmdfile=rman_duplicate.cmd
Recovery Manager: Release 19.0.0.0.0 - Production on Wed Nov 8 17:57:58 2023 Version 19.20.0.0.0
Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.
connected to auxiliary database: ECOM (not mounted)
RMAN> set encryption on identified by *; 2> set decryption identified by *
3> run (
4>   allocate      auxiliary      channel ch1 device      type      disk;
5>   allocate      auxiliary      channel ch2 device      type      disk;
6>   allocate      auxiliary      channel ch3 device      type      disk;
.
.
.
18>  allocate      auxiliary      channel ch15 device     type      disk;

executing command: SET encryption
executing command: SET decryption
allocated channel: ch1
channel ch1: SID-15383 device type-DISK
allocated channel: ch2
channel ch2: SID-16922 device type-DISK
allocated channel: ch3
channel ch3: SID-18459 device type-DISK
.
.
.
allocated channel: ch15
channel ch15: SID-20456 device type=DISK

Starting Duplicate Db at 08-DEC-23 searching for database ID
found backup of database ID 3935091107
... (command output omitted)

datafile 18 switched to datafile copy
input datafile copy RECID-15 STAMP=1152387334 file
name.+DATA/ECOM/086AD287EC980110E063090343822798/DATAFILE/ecom_stprt.265.1152381531 datafile 19
switched to datafile copy
input datafile copy RECID-16 STAMP=1152387334 file
name.+DATA/ECOM/086AD287EC980110E063090343822798/DATAFILE/ecom_ttpprt.267.1152381531 datafile 20
switched to datafile copy
input datafile copy RECID-17 STAMP.1152387334 file
name.+DATA/ECOM/086AD287EC980110E063090343822798/DATAFILE/ecom_exprrt.273.1152381537 datafile 21
switched to datafile copy
input datafile copy RECID-18 STAMP=1152387334 file name=4-
DATA/ECOM/086AD287EC980110E063090343822798/DATAFILE/ecom_ltprt.277.1152381631
contents of Memory Script:
Alter clone database open resetlogs; executing Memory Script
database opened
contents of Memory Script:
{
sql clone "alter pluggable database all open"; executing Memory Script
sql statement: alter pluggable database all open finished Duplicate Db at 08-DEC-23
Recovery Manager complete. [END]
```

Option 2: RMAN Restore/Recover Database

This procedure establishes on Power Virtual Server an Oracle database from an initial inconsistent data set, then applies incremental backups and archived redo logs to create a recovered version of the database as of a specific point in time.

The following steps should be executed as the oracle user:

1. Set appropriate ORACLE_SID.

```
# export ORACLE_SID=ECOM
```

2. Start the Oracle instance in NOMOUNT mode, directly specifying the pfile to be used.

```
connect / as sysdba
startup nomount pfile='/backup/ora/initECOM.ora'
create spfile='/u01/app/oracle/dbs/spfileECOM.ora' from pfile='/backup/ora/initECOM.ora';
shutdown immediate
startup nomount
show parameter spfile
```

NAME	TYPE	VALUE
spfile	string	/u01/app/oracle/dbs/spfileECOM.ora

3. Determine for the next step the control file included in the option2 level0 backup

```
# ls /backup/rman/ecom_option2_lev0/lev0_cf_*
/backup/rman/ecom_option2_lev0/lev0_cf_c-3935091107-20240118-00
```

4. Modify the script below to reflect the correct control file and then execute the rman script to restore the DB from level 0 backup and also apply any archived redo logs included in that backup via “recover database” as the “restore database” does not apply archived redo logs. Note the use of the backup TAG to specify from which backup we want to restore. The RMAN catalog in the control file may list multiple backups.

restore_option2_lev0.man

```
connect target /
set ENCRYPTION ALGORITHM 'AES256' ;
SET DECRYPTION IDENTIFIED BY passw0rd;
run {
restore controlfile from '/backup/rman/ecom_option2_lev0/lev0_cf_c-3935091107-20240118-00';
alter database mount;
restore database from tag ECOM_LEV0;
recover database;
}
```

Explanation of script:

- i. The backup files are encrypted and RMAN requires the encryption password to be able to restore the database files.
 - a. `SET DECRYPTION IDENTIFIED BY passw0rd;`
 - ii. We restore the database control file from the level 0 backup which contains the RMAN catalog we need to map backup pieces to data files as well as the list of data files and their expected locations.
 - a. `restore controlfile from
'/backup/rman/ecom_option2_lev0/lev0_cf_c-
3935091107-20240118-00';`
 - iii. We then alter the database to “mount mode” which is required for the next step.
 - a. `alter database mount;`
 - iv. Using the TAG we assigned to the level 0 backup we instruct RMAN to restore that database from that backup.
 - a. `restore database from tag ECOM_LEV0;`
 - v. As final step we apply any in level 0 backup included archive logs to the restored database. Note that this does not open the database so that we can apply future incremental backups and/or archive logs to the database.
 - a. `recover database;`
5. Execute the RMAN script. Ensure that the database instance was started with “startup nomount”.

```
# rman @restore_option2_lev0.man
```

Sample output is extensive and not listed here.

6. After the RMAN incremental level 0 is restored, catalog the remainder of the incremental level 1 backupsets and archived logs in the target directory.

If multiple incremental backups are provided, this will be an iterative process.

- o For all incremental backups, **EXCEPT the last**, you can simply execute a “recover database” after the incremental backup set / sets were cataloged.

The final incremental backup requires that the recovery is only until a specific time to be able to open the database!

Important: The target database can remain in MOUNT state and have incremental level 1 and archived REDO logs applied continuously, as needed. Catalog new backup pieces and archived logs and recover until ready for final cutover. The command below assumes that all incremental backups are stored in directories under ‘/backup/rman’.

```
RMAN> catalog start with '/backup/rman'
```

Sample output, just showing backup files from the first incremental level 1 backup:

```
RMAN> catalog start with '/backup/rman' ;

searching for all files that match the pattern /backup/rman

List of Files Unknown to the Database
File Name: /backup/rman/ecom_option2_incl/ctl_c-3935091107-20231030-00
File Name: /backup/rman/ecom_option2_incl/incl_if2a4rin_47_1
File Name: /backup/rman/ecom_option2_incl/incl_lg2a4rin_48_1
File Name: /backup/rman/ecom_option2_incl/incl_ah2a4rin_49_1
File Name: /backup/rman/ecom_option2_incl/incl_1i2a4rin_50_1
File Name: /backup/rman/ecom_option2_incl/incl_1j2a4rin_51_1
File Name: /backup/rman/ecom_option2_incl/incl_1k2a4rin_52_1
File Name: /backup/rman/ecom_option2_incl/incl_1l2a4rin_53_1
```

... lines omitted...

```
File Name: /backup/rman/ecom_option2_incl/incl_3e2a795k_111_1
File Name: /backup/rman/ecom_option2_incl/incl_3g2a795l_112_1
File Name: /backup/rman/ecom_option2_incl/incl_3h2a7950_113_1
File Name: /backup/rman/ecom_option2_incl/incl_3l2a795p_117_1
File Name: /backup/rman/ecom_option2_incl/incl_3m2a795p_118_1

Do you really want to catalog the above files (enter YES or NO)? Yes
cataloging files...
cataloging done
```

7. For all incremental backups, EXCEPT the final, you can then execute in RMAN:

```
RMAN> recover database;
```

8. After cataloging the FINAL incremental backup pieces the recovery needs to be up to the date/time determine in the backup section for Option 2) and highlighted in red. After this final recover no further changes are expected to be applied to the database from a recovery perspective.

```
RMAN> alter session set nls_date_format='MM-DD-YYYY HH24:mi:ss' ;
RMAN> recover database until time '01-13-2024 11:52:12';
```

9. After all interim archived REDO logs and the final RMAN increment level 1 have been cataloged and recovered as described above, the database can be opened.

```
# sqlplus / as sysdba
SQL > alter database open resetlogs ;
```

This concludes the Restore/Recover approach to database migration to Power Virtual Server.

Appendix

Seagate Lyve Mobile Solution for Physical Data Transfer

Overview

Reference: <https://www.seagate.com/products/cloud/lyve-mass-storage-platform/>

Lyve™ Mobile from Seagate® is a high-capacity edge storage solution that enables businesses to aggregate, store, move, and activate their data. Scalable, modular, and vendor agnostic, this integrated solution eliminates network dependencies so you can transfer mass data sets in a fast, secure, and efficient manner. With on-demand consumption delivered as a service, you order and pay only for the devices you need, when you need them.

As applied to RMAN backup and restore, Lyve Mobile can be used as a local store for source system backup. The device can be detached and physically shipped to Seagate. When supplied with Cloud Object Storage credentials, Seagate will, upon receipt, transfer the contents of the device across the network to Cloud Object Storage that a customer allocates, in this case, as staging space for the backup files to be restored to a target PowerVS instance.

Product and Service Brouchure:

https://www.seagate.com/content/dam/seagate/migrated-assets/www-content/product-content/data-transport/shared/files/SC704_1-2102US-Lyve-Mobile-Family-Flyer-revised.pdf

Physical Devices and Associated Capacities:

<https://www.seagate.com/products/data-transport/>