Backup and Duplicate

Objectives

- After completing this lesson, you should be able to:
  - Back up a CDB
  - Back up a PDB
  - Duplicate an active PDB into an existing CDB
  - Duplicate a CDB as encrypted
  - Validate CDB and PDBs

Goals

- Back up CDB and PDBs independently:
  - ARCHIVELOG mode at CDB level
  - CDB backups and PDB backups: cold and hot backups
- Recover CDB or PDBs:
  - Instance failure: CDB level
  - Complete media recovery:
    - CDB or PDB tempfile
    - Controlfile / redo log file / CDB root essential datafile: CDB mounted
    - PDB datafile: PDB opened if non-essential datafile / PDB mounted if essential datafile
  - Incomplete media recovery: CDB mounted or PDB closed
  - Flashback database: CDB mounted or PDB closed
  - Flashback PDB using PDB snapshots

Syntax and Clauses in RMAN

```
$ export ORACLE_SID=cdb1
$ rman TARGET /          ←————|————→          $ rman TARGET jim@pdb1
```

- DATABASE keyword operates on all PDBs and CDB root or on only one PDB.

```
RMAN> BACKUP DATABASE;
RMAN> RECOVER DATABASE;
```

- PLUGGABLE DATABASE keywords operate on individual PDBs.

```
RMAN> BACKUP PLUGGABLE DATABASE hr_pdb, sales_pdb;
RMAN> RECOVER PLUGGABLE DATABASE hr_pdb;
```

- Back up, restore, recover the CDB root using CDB$ROOT keyword.
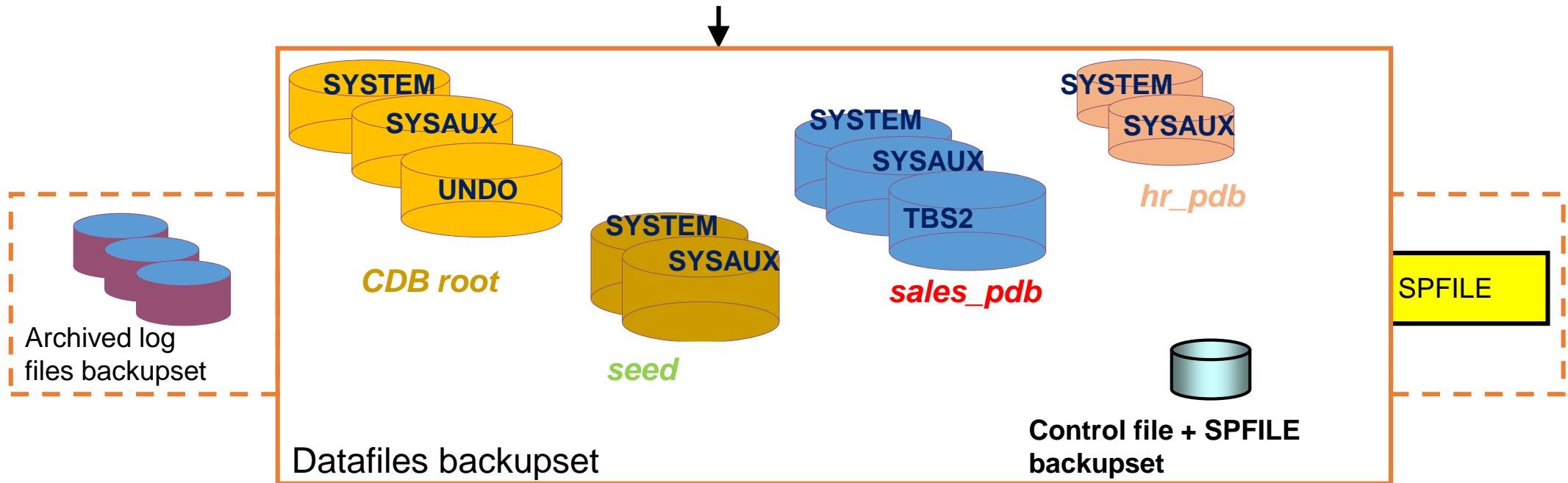
```
RMAN> BACKUP PLUGGABLE DATABASE "CDB$ROOT";
```

- Qualify tablespace of PDB with PDB name.

```
RMAN> BACKUP TABLESPACE sales_pdb:tbs2;
RMAN> RESTORE TABLESPACE system;
```

CDB Backup: Whole CDB Backup

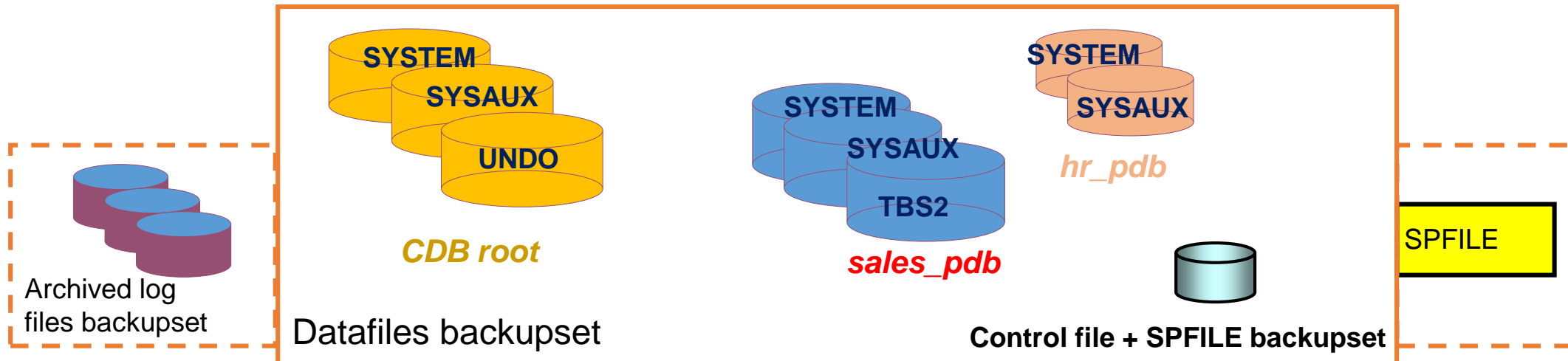- Back up all PDBs datafiles and CDB root files.

```
$ rman TARGET /
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;
RMAN> BACKUP DATABASE PLUS ARCHIVELOG;
```
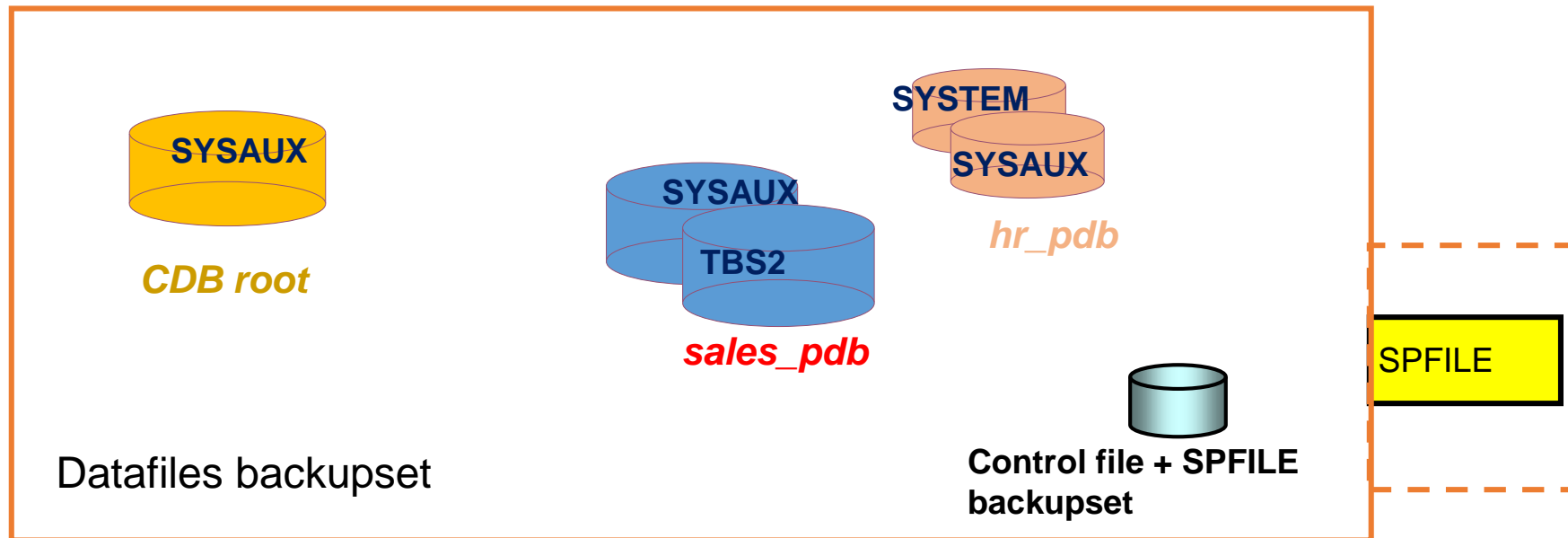
CDB Backup: Partial CDB Backup

```
$ rman TARGET /
RMAN> BACKUP PLUGGABLE DATABASE "CDB$ROOT", sales_pdb;
RMAN> BACKUP PLUGGABLE DATABASE hr_pdb PLUS ARCHIVELOG;
```

```
$ rman TARGET sys@hr_pdb
RMAN> BACKUP DATABASE;
```

PDB Backup: Partial PDB Backup

```
$ rman TARGET /
RMAN> REPORT SCHEMA;
RMAN> BACKUP TABLESPACE sales_pdb:tbs2;
RMAN> BACKUP TABLESPACE hr_pdb:system, sales_pdb:sysaux;
RMAN> BACKUP TABLESPACE sysaux, hr_pdb:sysaux;
```

Using RMAN Backup to Plug an Unplugged PDB

Duplicating Pluggable Databases

- A single pluggable database

```
RMAN> DUPLICATE DATABASE TO cdb1 PLUGGABLE DATABASE pdb1;
```

- Several pluggable databases

```
RMAN> DUPLICATE DATABASE TO cdb1 PLUGGABLE DATABASE pdb1, pdb3;
```

- All pluggable databases except one

```
RMAN> DUPLICATE DATABASE TO cdb1 SKIP PLUGGABLE DATABASE pdb3;
```

- A PDB and tablespaces of other PDBs

```
RMAN> DUPLICATE DATABASE TO cdb1
               PLUGGABLE DATABASE pdb1 TABLESPACE pdb2:users;
```

Cloning Active PDB into an Existing CDB

Duplicate a PDB or PDB tablespaces in active mode to an existing opened CDB.

- Set the `COMPATIBLE` initialization parameter to 18.1.

- Clone only one PDB at a time.

  - Set the destination CDB in RW mode.

  - Set the `REMOTE_RECOVERY_FILE_DEST` initialization parameter in the destination CDB to the location where to restore foreign archive log files.

```
RMAN> DUPLICATE PLUGGABLE DATABASE pdb1 AS pdb2 FROM ACTIVE DATABASE
                        DB_FILE_NAME_CONVERT ('cdb1', 'cdb2');
```

Example: 1

To duplicate `pdb1` from `CDB1` into `CDB2`:

1. Set the `REMOTE RECOVERY FILE DEST` initialization parameter in `CDB2`.

```
SQL> ALTER SYSTEM SET REMOTE_RECOVERY_FILE_DEST='/dir_to_restore_archive log files';
```

2. Connect to the source (`TARGET` for `DUPLICATE` command): `CDB1`

3. Connect to the existing `CDB2` that acts as the auxiliary instance:

```
RMAN> CONNECT TARGET "sys/oracle_4U@cdb1 AS SYSDBA"
RMAN> CONNECT AUXILIARY "sys/oracle_4U@cdb2 AS SYSDBA"
```


CDB1 — pdb1   **DUPLICATE pdb1** →   CDB2 — pdb1

4. Start duplicate.

```
RMAN> DUPLICATE PLUGGABLE DATABASE pdb1 TO cdb2 FROM ACTIVE DATABASE;
```

Example: 2

To duplicate `pdb1` from `CDB1` into `CDB2 as pdb2`:

1. Set the `REMOTE_RECOVERY_FILE_DEST` initialization parameter in `CDB2`.

```
SQL> ALTER SYSTEM SET REMOTE_RECOVERY_FILE_DEST='/dir_to_restore_archive log files';
```

2. Connect to the source (`TARGET` for `DUPLICATE` command): `CDB1`

3. Connect to the existing `CDB2` that acts as the auxiliary instance:

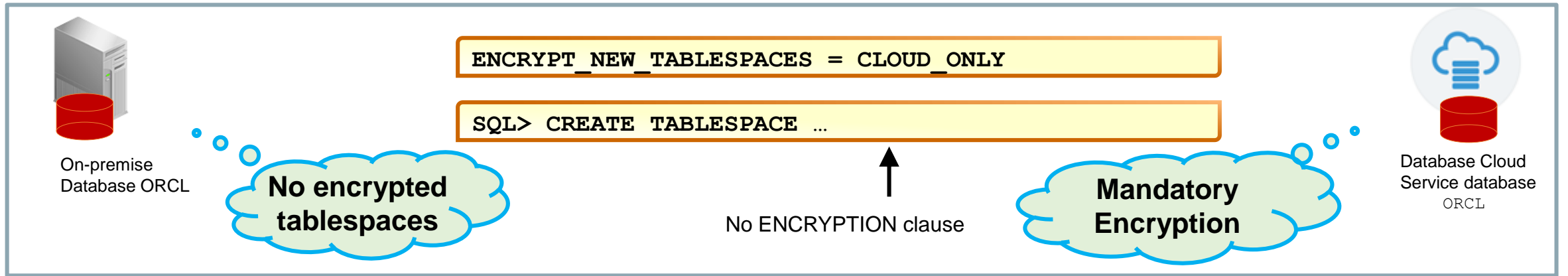```
rman TARGET sys@cdb1 AUXILIARY sys@cdb2
```



4. Start duplicate.

```
RMAN> DUPLICATE PLUGGABLE DATABASE pdb1 AS pdb2 TO cdb2 FROM ACTIVE DATABASE;
```

Duplicating On-Premise CDB as Cloud Encrypted CDB

- Duplicating an on-premise CDB to the Cloud:
  - Any newly created tablespace is encrypted in the Cloud CDB.


```
ENCRYPT_NEW_TABLESPACES = CLOUD_ONLY
```

```
SQL> CREATE TABLESPACE …
```

On-premise
Database ORCL

No encrypted tablespaces

No ENCRYPTION clause

Mandatory Encryption

Database Cloud
Service database
ORCL

- The Cloud CDB holds a keystore because this is the default behavior on Cloud.
- All forms of normal duplication are compatible:
  - Active duplication
  - Backup-based duplication
  - Targetless duplicate

Duplicating On-Premise Encrypted CDB as Cloud Encrypted CDB

Duplicating an on-premise CDB with encrypted tablespaces to the Cloud:

1. Tablespaces of the source CDB need to be decrypted.
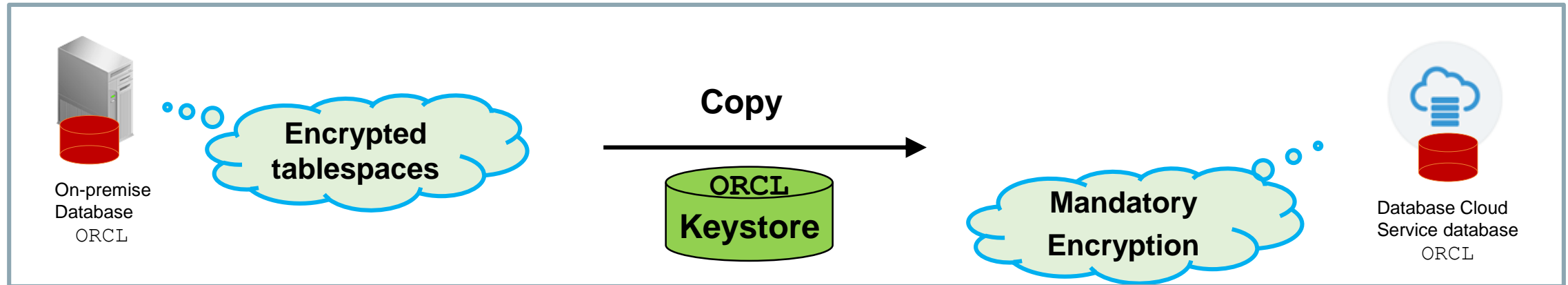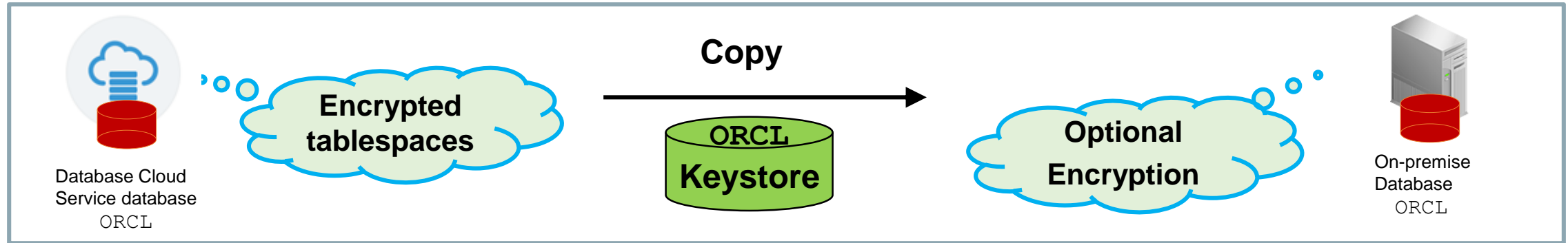

2. Restored tablespaces are re-encrypted in the Cloud CDB.
   - Requires the master TDE key from the source CDB keystore
   - Requires the source keystore to be copied and opened at the destination CDB

```
RMAN> SET DECRYPTION WALLET OPEN IDENTIFIED BY password;
RMAN> DUPLICATE DATABASE TO orcl FROM ACTIVE DATABASE AS ENCRYPTED;
```

Migrating Cloud Encrypted CDB as On-Premise CDB

1. Tablespaces of the source CDB are necessarily encrypted.

2. Restored tablespaces need to be decrypted to be created:
   - Requires the TDE master key from the source CDB keystore
   - Requires the source keystore to be copied and opened at the destination CDB

```
RMAN> SET DECRYPTION WALLET OPEN IDENTIFIED BY password;
RMAN> DUPLICATE DATABASE TO orcl FROM ACTIVE DATABASE AS DECRYPTED;
```

Checking for Block Corruption

- Invoking proactive health check of the database and its components using RMAN `VALIDATE` command:
  - Scans the specified files and verifies their contents
    - CDB: All datafiles of the CDB root and PDBs

```
RMAN> VALIDATE DATABASE;
```

    - CDB root: All datafiles of the CDB root only

```
RMAN> VALIDATE DATABASE ROOT;
```

    - PDB: All datafiles of the listed PDBs

```
RMAN> VALIDATE PLUGGABLE DATABASE pdb1, pdb2;
```

  - Confirms that the datafiles exist and are in the correct location
  - Checks for corrupt data blocks

Summary

- In this lesson, you should have learned how to:
    - Back up a CDB
    - Back up a PDB
    - Duplicate an active PDB into an existing CDB
    - Duplicate a CDB as encrypted
    - Validate CDB and PDBs

Practice 8: Overview

- 8-1: RMAN whole CDB backup
- 8-2: RMAN PDB backup
- 8-3: Duplicating a PDB into an existing CDB
- 8-4: Duplicating an on-premises CDB for Cloud