Managing Data Files

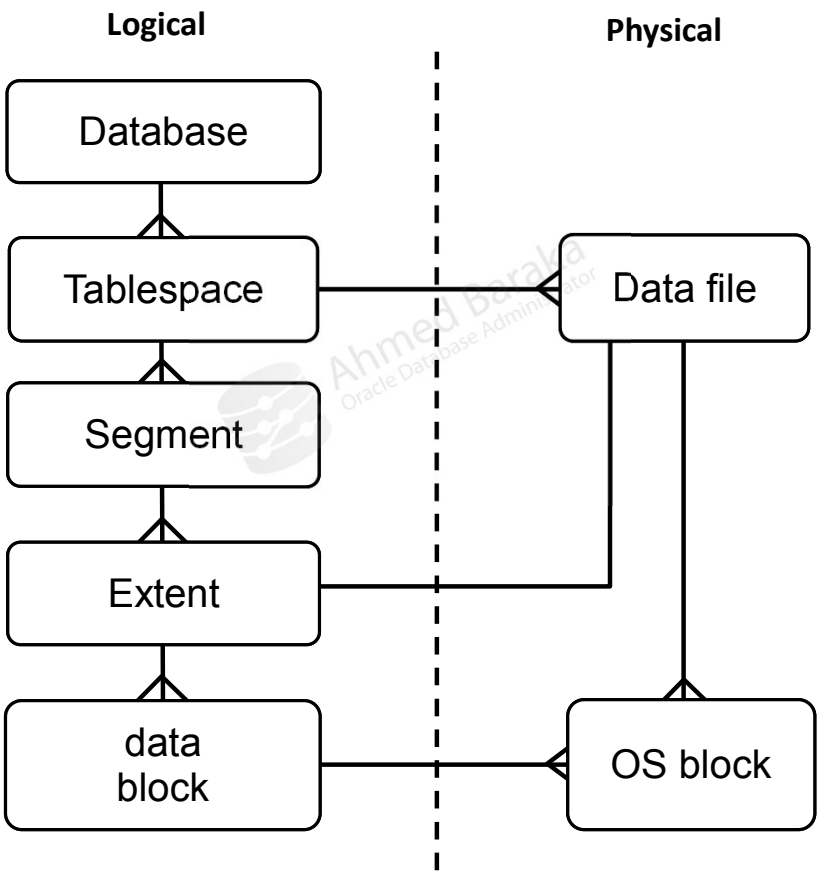
By Ahmed Baraka

Objectives

In this lecture, you will learn how to perform the following:

- Resize data files
- Alter data file availability
- Rename and relocate data files
- Drop a datafile

Logical and Physical Database Structures



Manually Resizing Data Files

- Data files can be increased/decreased:

```
ALTER DATABASE DATAFILE '<data file>' RESIZE <n>M|G|T;
```

- Example:

```
ALTER DATABASE DATAFILE
'/u01/oracle/oradata/ORADB/datafile/o1_mf_system_k7.dbf'
RESIZE 100M;
```

- May be needed in the following scenarios:
  - Fix an exaggerated tablespace estimation
  - Increase a datafile size instead of adding a new data file
  - Prepare for a planned large data loading

Altering Data File Availability

- Making a datafile offline is needed in the following scenarios:
  - You want to rename or relocate a data file (not mandatory)
  - Take an offline backup for the datafile
  - A data file becomes missing or corrupted. You must take it offline before you can open the database
- Syntax:

```
ALTER DATABASE DATAFILE '<datafile>' ONLINE | OFFLINE [FOR DROP];

-- it affects all the data/temp files in the tablespace
ALTER TABLESPACE ... DATAFILE {ONLINE|OFFLINE}
ALTER TABLESPACE ... TEMPFILE {ONLINE|OFFLINE}
```

  - **FOR DROP** is used when the database is running in **NOARCHIVELOG** mode. It marks the data file to be dropped and cannot be brought online

Renaming and Relocating Online Data Files

- You need to rename or relocate data files when you want to move the data files from one type of storage to another or from one location to another (including ASM diskgroups)
- Online rename and relocation mean the data file is physically renamed or moved while users are accessing the datafile
- Syntax:

```
ALTER DATABASE
 MOVE DATAFILE '<old name>' TO 'new name' [REUSE] [KEEP];
```

  - **REUSE** is used to overwrite an existing file
  - **KEEP** is used to retain the data file in the old location and copy it to the new location

Renaming and Relocating Online Data Files: Examples

- To rename a data file in the same location:

```
ALTER DATABASE MOVE DATAFILE '/u01/oracle/rbdb1/user1.dbf'
TO '/u01/oracle/rbdb1/user01.dbf';
```

- To move a data file from one location to another:

```
ALTER DATABASE MOVE DATAFILE '/u01/oracle/rbdb1/user1.dbf'
TO '/u02/oracle/rbdb1/user1.dbf';
```

- To copy a data file from one location to another:

```
ALTER DATABASE MOVE DATAFILE '/u01/oracle/rbdb1/user1.dbf'
TO '/u02/oracle/rbdb1/user1.dbf' KEEP;
```

Renaming and Relocating Offline Data Files

- The data files to be renamed or relocated are *cleanly* offline

1. Make sure the data file is offline
2. Relocate or rename the data files externally (e.g. using OS commands)
3. Issue either of the following statements:

```
ALTER DATABASE RENAME FILE '..' TO '..';
ALTER TABLESPACE <tbs-name> RENAME DATAFILE '..' TO '..';
```

4. (If the DB is open) Apply the most recent redo logs:

```
RECOVER DATAFILE '...';
```

5. Take backup of the database

Dropping Data Files

- The data file must be empty (not segment extent in it)
- Data file dropped from data dictionary view, control files, and physically
- The statement syntax:

```
ALTER TABLESPACE <tbs-name> DROP DATAFILE | TEMPFILE '...';
ALTER DATABASE TEMPFILE '...' DROP INCLUDING DATAFILES;
```

  - The database must be open
  - The only data file in a tablespace cannot be dropped. Drop the tablespace including datafiles instead.

Summary

In this lecture, you should have learnt how to perform the following:

- Resize data files
- Alter data file availability
- Rename and relocate data files
- Drop a datafile