

Managing the Redo Log

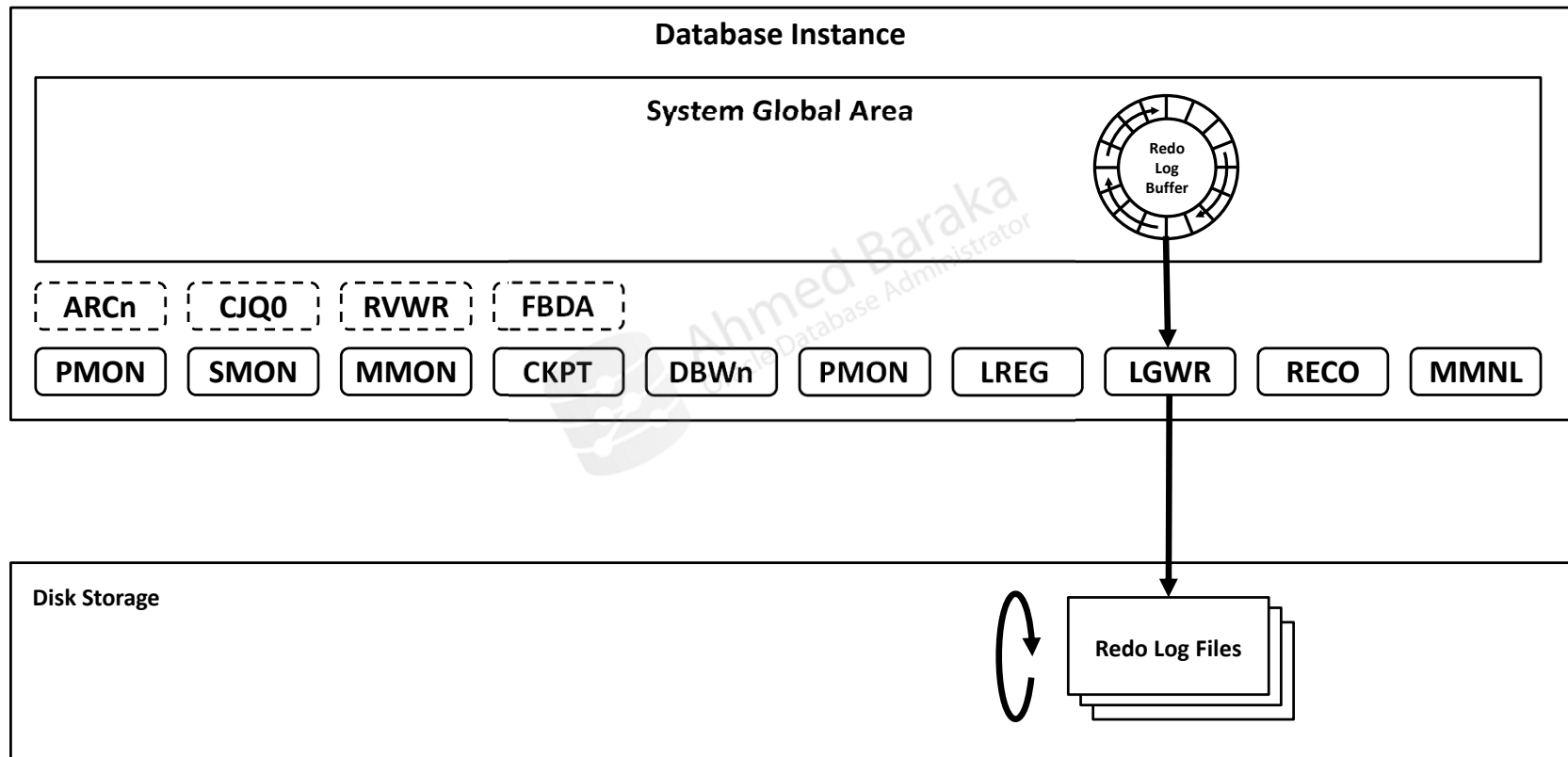
By Ahmed Baraka

Objectives

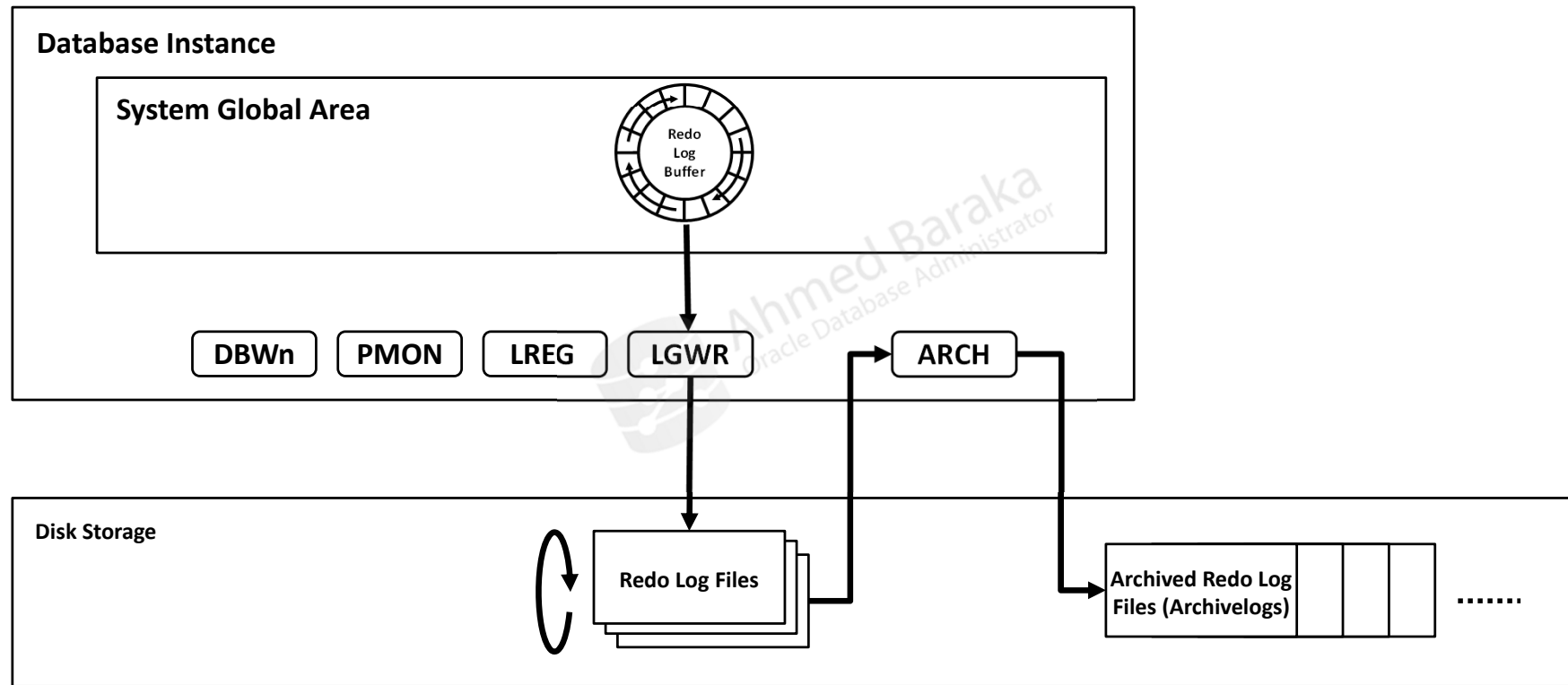
In this lecture, you should learn how to perform the following:

- Describe using redo log files in NOARCHIVELOG and ARCHIVELOG modes
- Describe the cycle of writing into redo log files
- Understand the difference between clean shutdown and abnormal shutdown
- Describe instance recovery mechanism
- Manage redo log groups and members
- Compare between undo and redo data
- Understand the best practices of managing redo log

DB Architecture in NOARCHIVELOG Mode



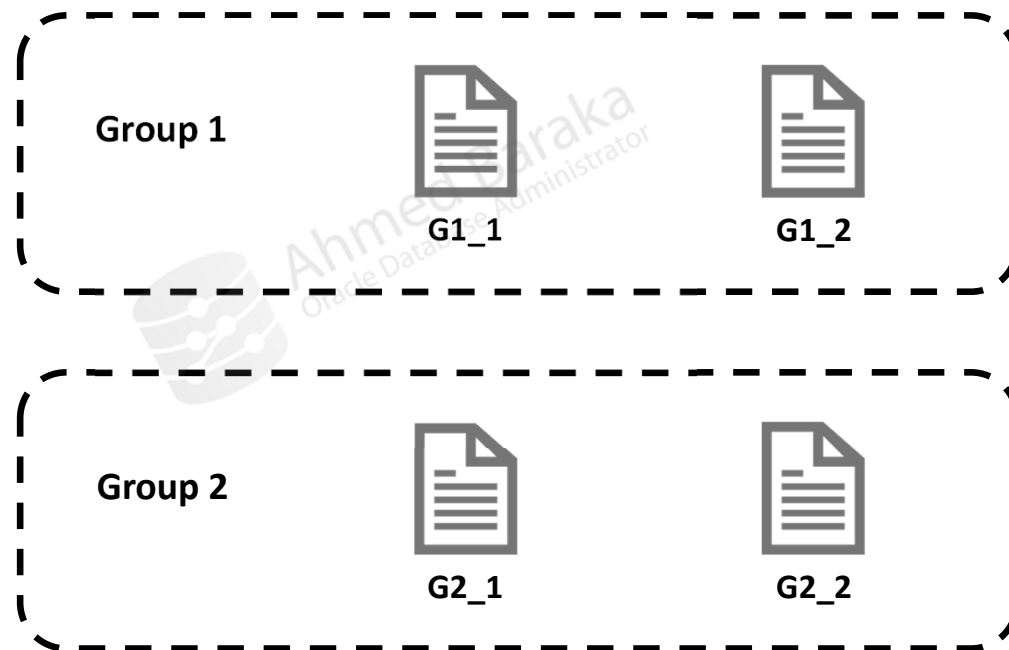
DB Architecture in ARCHIVELOG Mode



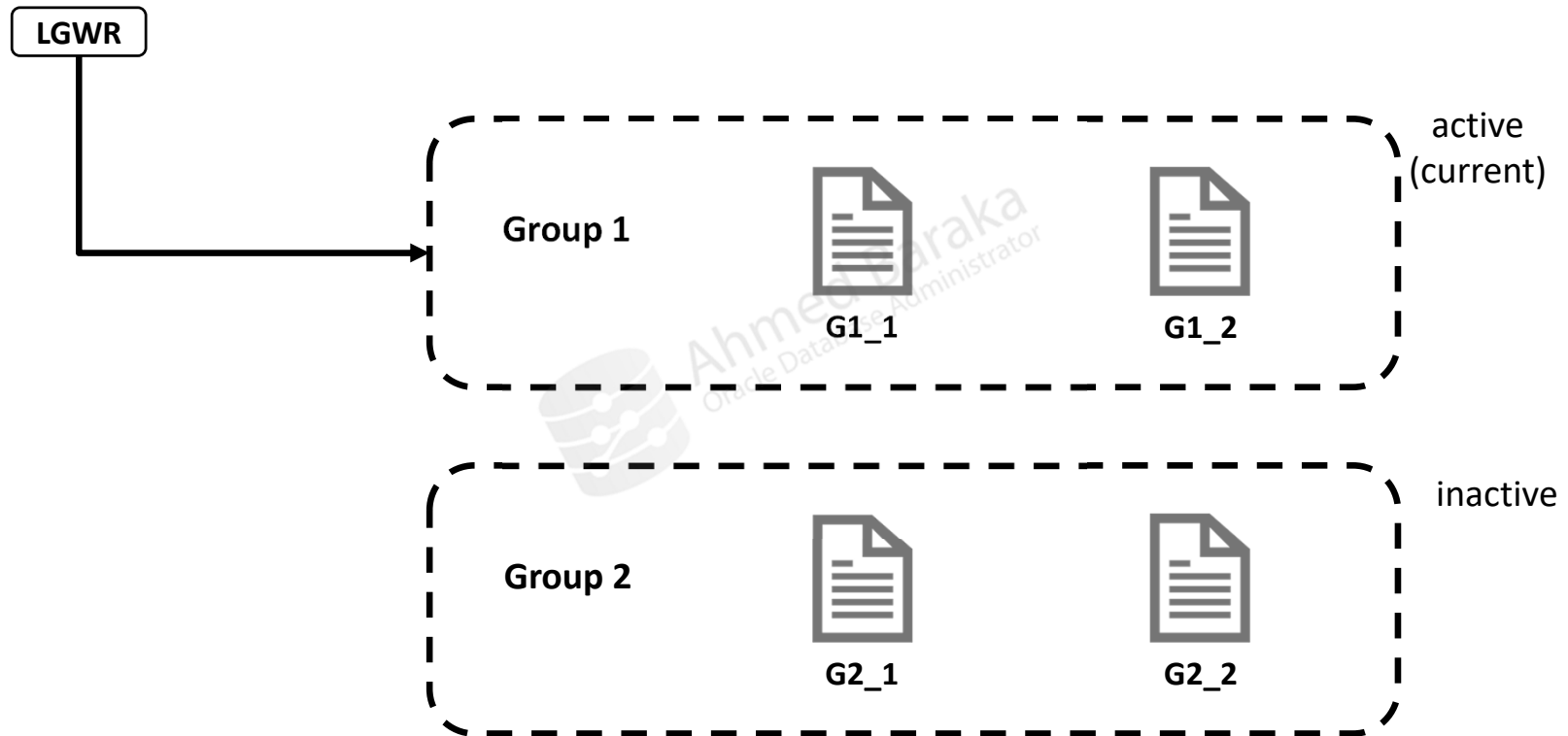
About NOARCHIVELOG and ARCHIVELOG Modes

- In either modes, the redo entries are handled as follows:
 - LOGWR processes saves the redo log entries from the **log buffer cache** area into the **online redo log groups** in a circular fashion
 - Online redo log file are used for **instance recovery**
 - Online redo log files are needed for normal database operations
- In **ARCHIVELOG** mode, the archiver (ARCH) process reads the redo log entries from the online redo log groups into archived redo log files.
 - Archived redo log files are generated **sequentially**
 - Archived redo log files are not needed by the database normal operations. They are used only for recoverability purposes

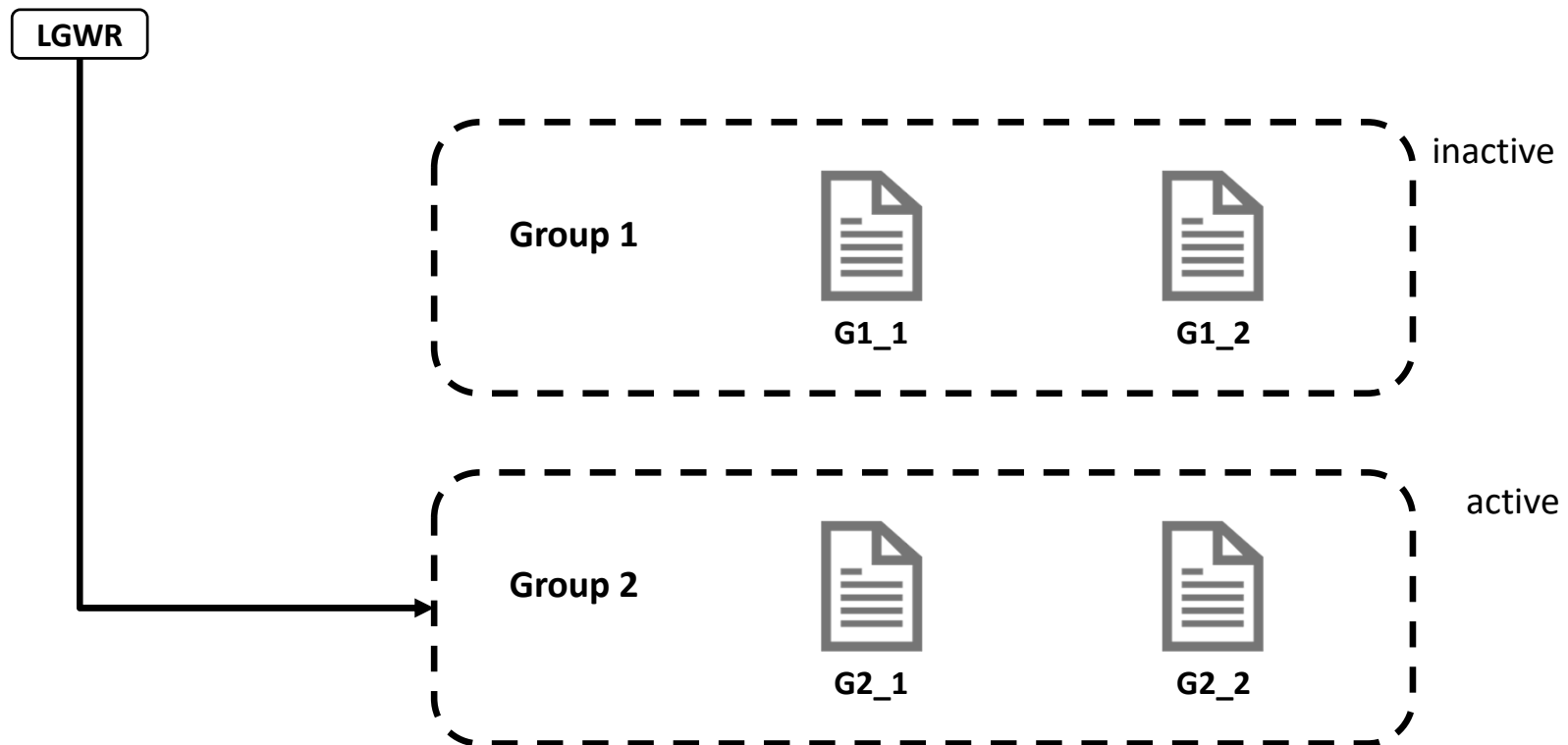
Writing into Redo Log Cycle



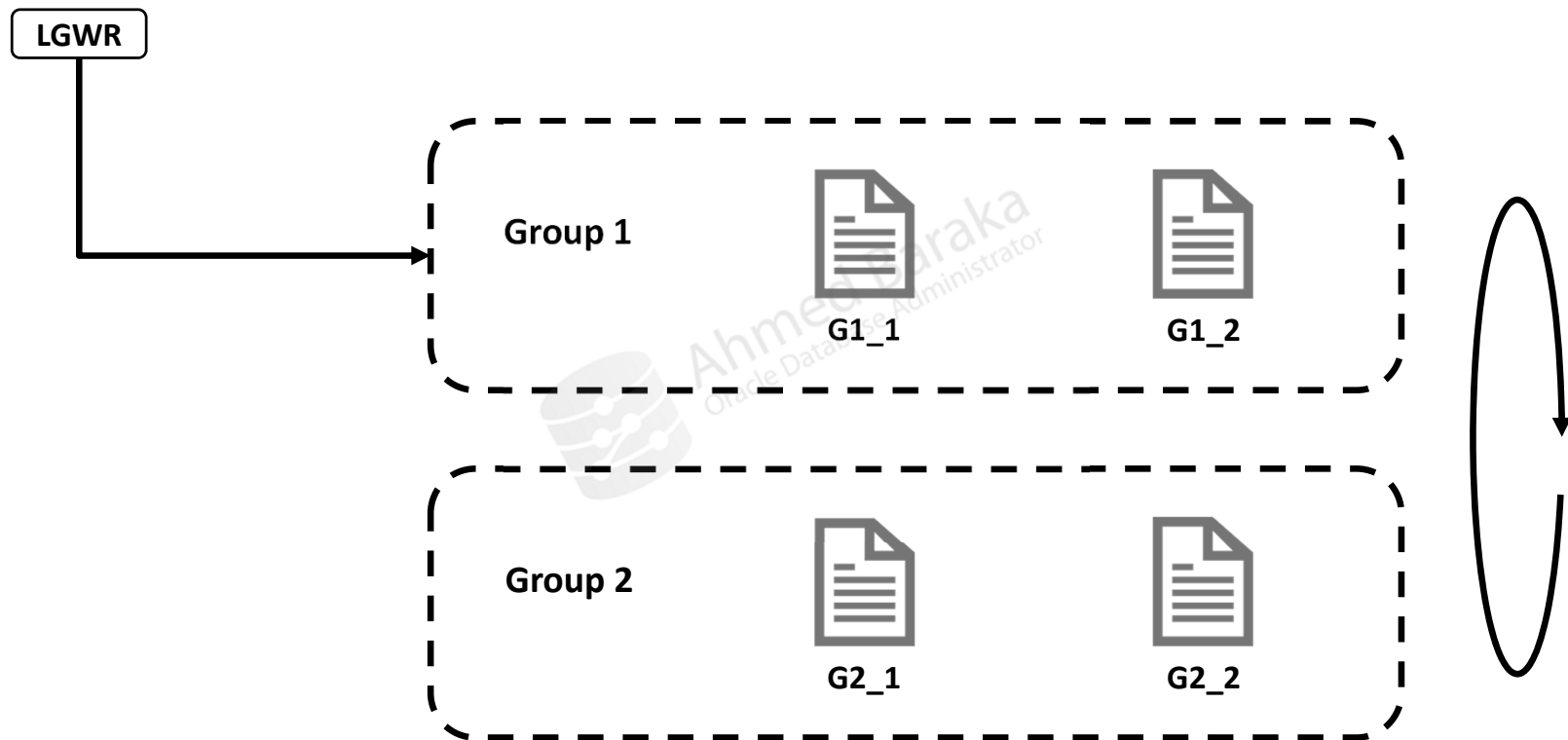
Writing into Redo Log Cycle



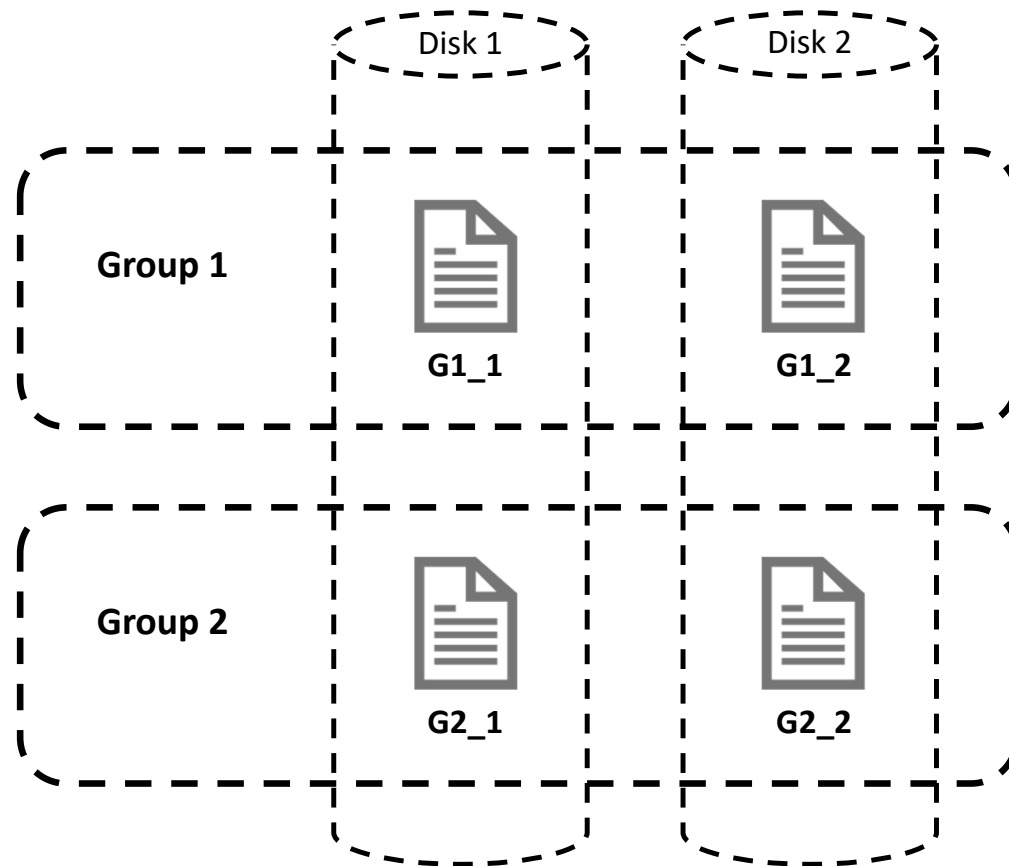
Writing into Redo Log Cycle



Writing into Redo Log Cycle



Redo Log Multiplexing

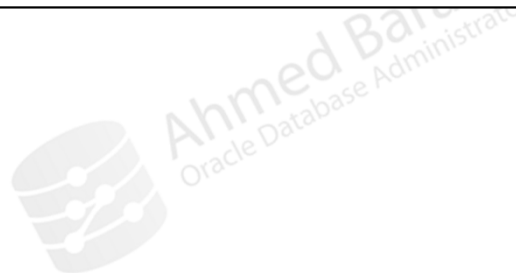


About Redo Log Files

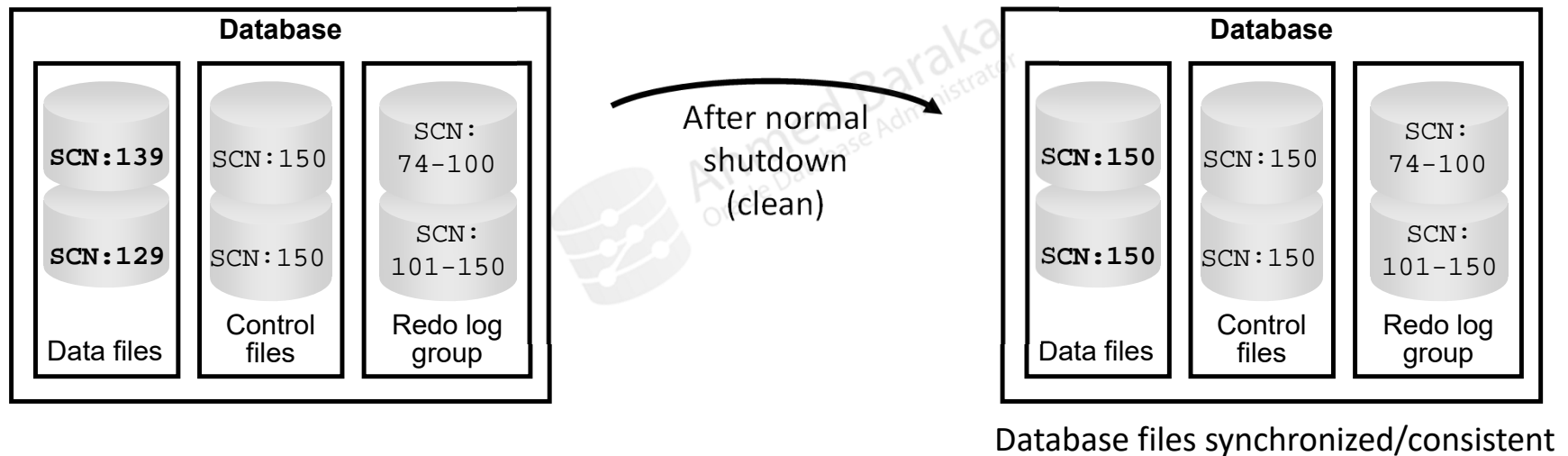
- LGWR writes to redo log files in a circular fashion
- Log files are multiplexed into **groups**: identical redo log files in a group
 - At least two groups must be there. dbca creates a database with 3 groups
- Redo log files in a group are called **members**
 - All members are of the same size
 - At least one member must exist in each group
- The group that is being accessed by the LGWR is **active** or **current**
- **Redo Log switch** is the process when LGWR stops writing into the current group and starts writing into the next group
- Redo log multiplexing provides a level of redundancy to improve the database availability

Obtaining Information About the Online Redo Log

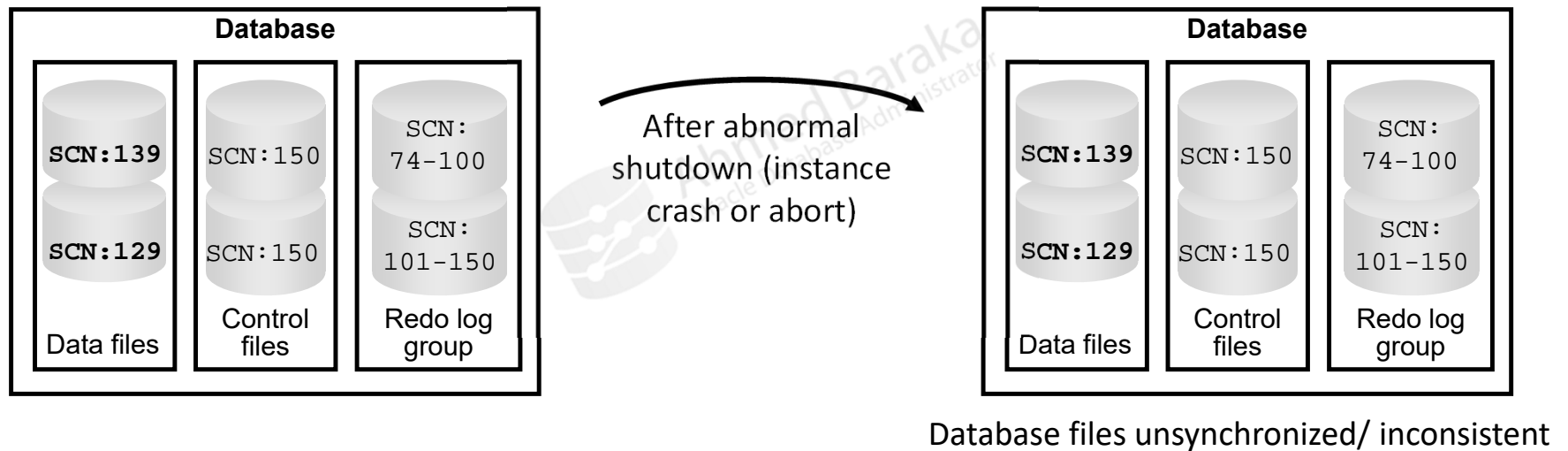
View	Description
V\$LOG	Displays the redo log file information from the control file
V\$LOGFILE	Retrieves redo log groups and members and member status



Database Datafiles Synchronization After Normal Shutdown



Database Datafiles Synchronization After Abnormal Shutdown



Instance Recovery

- Is caused by attempts to open a database whose data files are not synchronized on shutdown
- Recover procedure:
 - **Rolling forward:** read the redo log groups to roll the data files forward to the time of shutdown
 - Opening the database (without allowing the users to connect)
 - **Rolling back:** changes that are made but not committed are returned to their original state (using undo data)

Managing Redo Log Groups and Members

- Create new redo log groups
 - To create a new group that joins the existing groups: the new member size must be the same as the existing member size
 - To be part of the groups that will replace the existing groups: the new members could have different sizes
- Create new redo log members
- Relocate and rename redo log members
- Drop redo log groups and members

Creating Redo Log Groups and Members

- A database can have up to **MAXLOGFILES** groups.

```
SELECT RECORDS_TOTAL as MAXLOGFILES FROM  
V$CONTROLFILE_RECORD_SECTION WHERE type = 'REDO LOG';
```

- Make sure that eventually all the redo members have the same size
- Submit the following statement (OMF is supported):

```
ALTER DATABASE  
ADD LOGFILE ('/u01/oracle/dbs/log1c.rdo',  
'/u01/oracle/dbs/log2c.rdo') SIZE 1024M;
```

- Group number can be specified:

```
ALTER DATABASE  
ADD LOGFILE GROUP 4  
( '/u01/oracle/dbs/log1c.rdo', '/u01/oracle/dbs/log2c.rdo' ) SIZE 100M;
```

Creating Redo Log Groups in OMF

- Redo log group member locations obey the OMF of datafiles
`DB_CREATE_FILE_DEST`
- We can set an OMF specifically for the redo log groups (and control files) by setting the following parameters:
 - `DB_CREATE_ONLINE_LOG_DEST_1 = '/u01/oradata'`
 - `DB_CREATE_ONLINE_LOG_DEST_2 = '/u02/oradata'`
- If OMF is enabled, we can create a redo log group as follows:

```
ALTER DATABASE ADD LOGFILE SIZE 1024M;
```

Creating Redo Log Members in SQL

- The group must not be the **CURRENT**

```
SELECT GROUP#, STATUS FROM V$LOG ORDER BY 1;
```

- To make the new group as the current group, perform log switch:

```
ALTER SYSTEM SWITCH LOGFILE;
```

- Submit the following statement:

```
ALTER DATABASE ADD LOGFILE MEMBER  
'/u01/oracle/dbs/log2c.rdo' TO GROUP 2;
```

- You do not have to set the size (it automatically take the current size)
- File name must be provided even if the OMF is enabled
- You cannot add a member to a group if all the group members have been lost or deleted

Creating Redo Log Groups and Members in dbca

Database Configuration Assistant - Create 'orcl' database - Step 12 of 15

Select Database Creation Option

19c ORACLE Database

Select the database creation options.

☐ Create database

Specify the SQL scripts you want to run after the database is created. The scripts are run in the order listed below.

Post DB creation scripts: Browse...

☐ Save as a database template

Template name:

Template location: Browse...

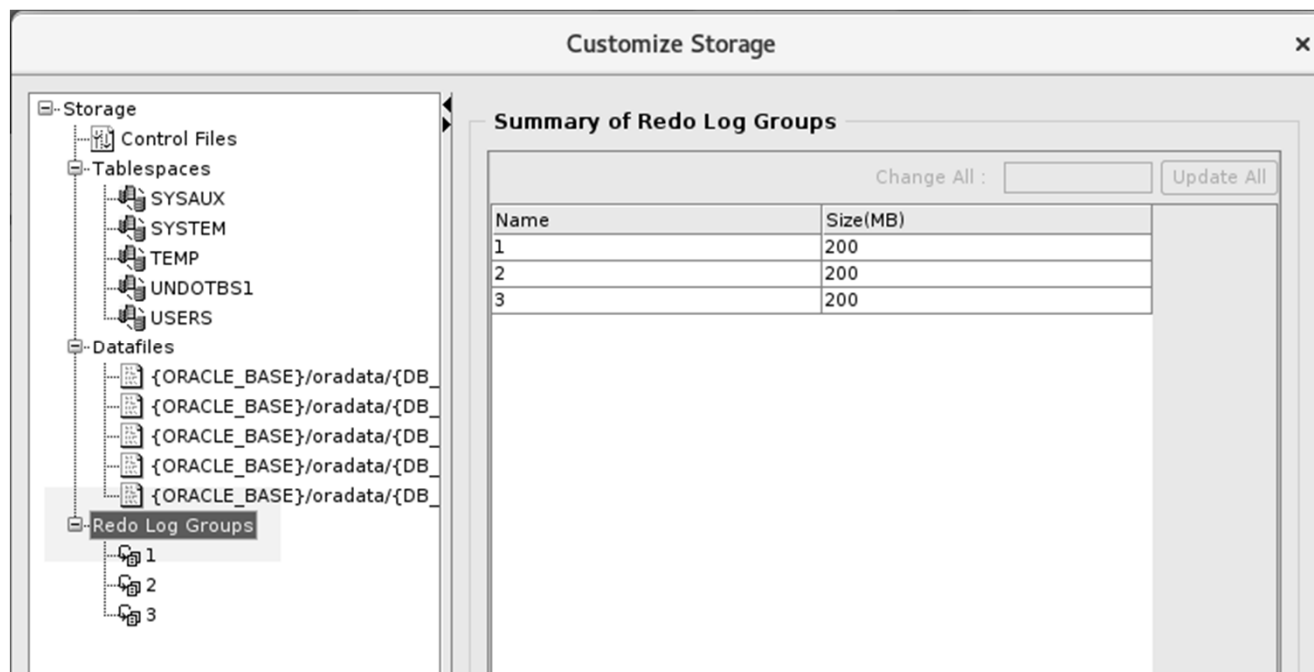
Description:

☐ Generate database creation scripts

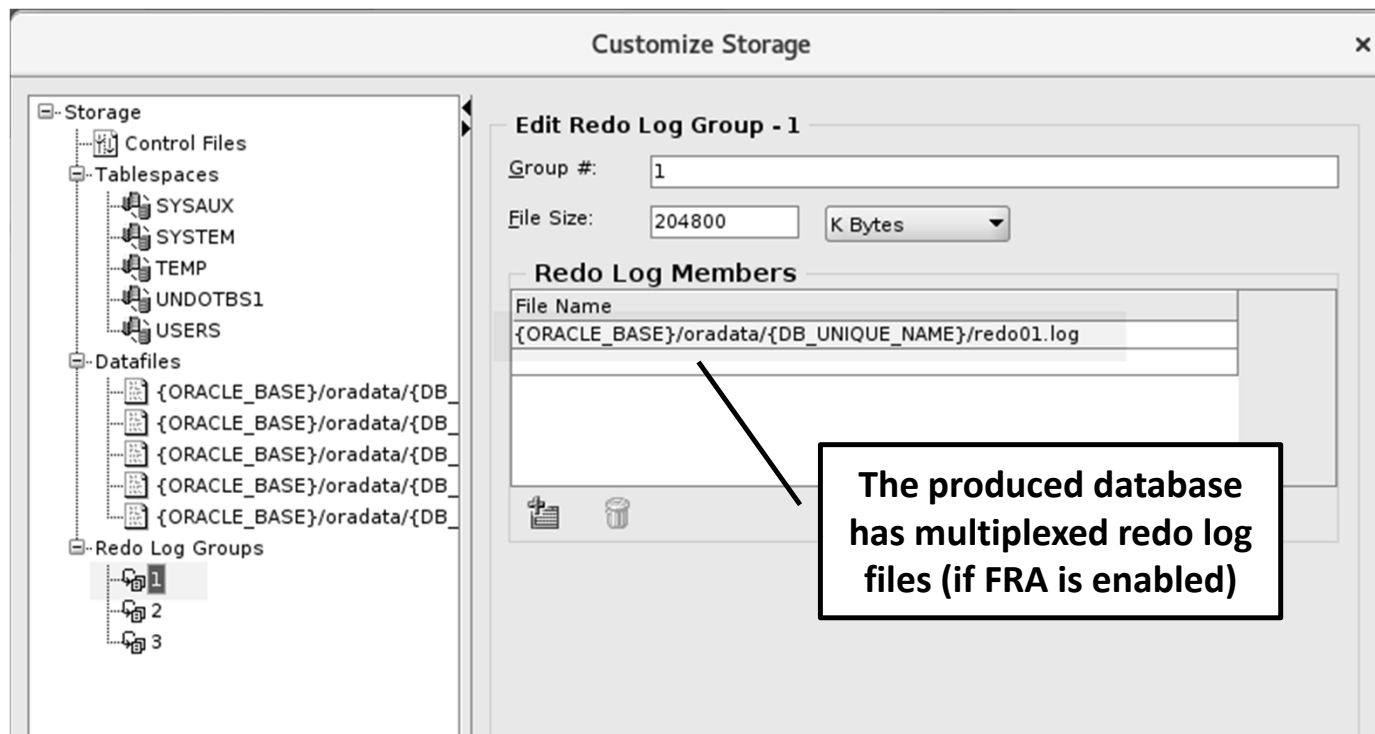
Destination directory: Browse...

Following advanced configuration options can be used to configure initialization parameters and customize database storage locations.

Creating Redo Log Groups and Members in dbca



Creating Redo Log Groups and Members in dbca



Relocating and Renaming Redo Log Members

1. Shutdown the database
2. Copy the redo log files to the new location
3. Mount the database
4. Submit the following statement to rename the redo log members:

```
ALTER DATABASE  
RENAME FILE '/u01/logs/log1a.rdo', '/u01/logs/log2a.rdo'  
TO '/u02/logs/log1c.rdo', '/u02/logs/log2c.rdo';
```

5. Startup the database in normal mode

Dropping Redo Log Groups and Members

- The database requires at least two redo log groups to operate
- You can drop a redo log group only if it is inactive (and archived)
- Submit the following statement to drop a group and its members:

```
ALTER DATABASE DROP LOGFILE GROUP 3;
```

- Submit the following statement to drop a member:

```
ALTER DATABASE DROP LOGFILE MEMBER '/u01/oracle/dbs/log3c.rdo';
```


Forcing Log Switches

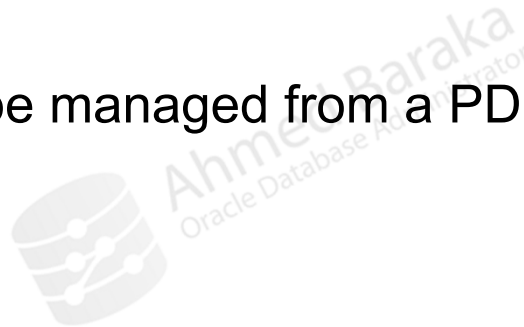
- Submit the following statement:

```
ALTER SYSTEM SWITCH LOGFILE;
```



Redo Log in Multitenant Environment

- Redo logs are linked to the CDB and shared by all the PDBs
- Multitenant environments hosting multiple PDBs usually require large redo log groups
- Redo log files cannot be managed from a PDB



Comparison Between Undo and Redo Data

	Undo	Redo
Target	How to undo a change	How to reproduce a change
Used for	Rollback (undo changes) Read consistency Flashback	Data recovery Instance crash recovery (roll forward) Replication
Stored in	Undo tablespaces (segments)	Online redo log Archived log files
Protects against	inconsistence reads instance crash	data loss instance crash

Managing the Redo Log: Best Practices

- Have at least 3 redo log groups
- If multiple storage mounts (disks, Disk Group, LUN) are attached, multiplex every group members into different disks
- If saved in a single storage mount, it should support stripping and mirroring (RAID or ASM)
- Make the redo group members are large enough
 - Big enough so that log switches are performed not sooner than 15 minutes
- Locate them in the storage with lowest latency
- Almost all production systems operate in ARCHIVELOG mode

Summary

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

- Describe using redo log files in NOARCHIVELOG and ARCHIVELOG modes
- Describe the cycle of writing into redo log files
- Understand the difference between clean shutdown and abnormal shutdown
- Describe instance recovery mechanism
- Manage redo log groups and members
- Compare between undo and redo data
- Understand the best practices of managing redo log