



# Oracle 11g DBA Fundamentals Overview

Lesson 06: Managing Tablespaces



# Lesson Objectives

- Creating Tablespaces
- Altering Tablespace Availability
- Using Read-Only Tablespaces
- Renaming Tablespaces
- Managing the SYSAUX Tablespace





# Table spaces

A tablespace is a logical storage unit within an Oracle database.

It is logical because a table space is not visible in the file system of the machine on which the database resides.

A table space, in turn, consists of at least one data file which, in turn, are physically located in the file system of the server.

A datafile belongs to exactly one tablespace.

Each table, index and so on that is stored in an Oracle database belongs to a table space.

The table space builds the bridge between the Oracle database and the file system in which the table's or index' data is stored.

There are three types of table spaces in Oracle:

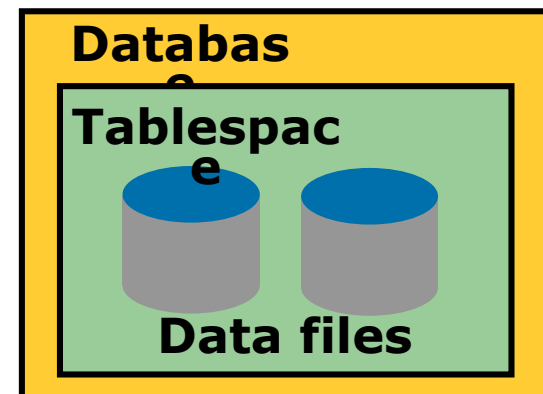
- Permanent table spaces
- Undo table spaces
- temporary table spaces



# Tablespaces and Data Files

Oracle stores data logically in tablespaces and physically in data files.

- Tablespaces:
  - Can belong to only one database at a time
  - Consist of one or more data files
  - Are further divided into logical units of storage
- Data files:
  - Can belong to only one tablespace and one database
  - Are a repository for schema object data





# Space Management in Tablespaces

- Locally managed tablespace:
  - Free extents are managed in the tablespace.
  - Bitmap is used to record free extents.
  - Each bit corresponds to a block or group of blocks.
  - Bit value indicates free or used.
- Dictionary-managed tablespace:
  - Free extents are managed by the data dictionary.
  - Appropriate tables are updated when extents are allocated or deallocated.

## Extent Management

- ☒ Locally Managed
- ☐ Dictionary Managed



# Creating a New Tablespace

Database: [orcl.oracle.com](#) > [Tablespaces](#) > Create Tablespace Logged in As SYS

## Create Tablespace

[Show SQL](#) [Cancel](#) [OK](#)

**General** [Storage](#) [Thresholds](#)

\* Name

**Extent Management**

- ☒ Locally Managed
- ☐ Dictionary Managed

**Type**

- ☒ Permanent
  - ☐ Set as default permanent tablespace
- ☐ Temporary
  - ☐ Set as default temporary tablespace
- ☐ Undo

**Status**

- ☒ Read Write
- ☐ Read Only
- ☐ Offline

**Datafiles**

☐ Use bigfile tablespace  
Tablespace can have only one datafile with no practical size limit.

[Add](#)

<a href="#">Edit</a> <a href="#">Remove</a>			
Select	Name	Directory	Size (MB)
<input checked="" type="radio"/>	<a href="#">inventory01.dbf</a>	/u01/app/oracle/oradata/orcl/	50.00



# Storage for Locally Managed Tablespaces

Database: [orcl.us.oracle.com](#) > [Tablespaces](#) > Create Tablespace Logged in As SYS

## Create Tablespace

Show SQL Cancel OK

[General](#) **Storage** [Thresholds](#)

### Extent Allocation

☒ Automatic

☐ Uniform

Size  KB

### Segment Space Management

☒ Automatic

Objects in the tablespace automatically manage their free space. It offers high performance for free space management.

☐ Manual

Objects in the tablespace will manage their free space using free lists. It is provided for backward compatibility.



# Tablespaces in the Preconfigured Database

SYSTEM  
SYSAUX  
TEMP

- UNDOTBS1
- USERS
- EXAMPLE

									Create
									Edit View Delete Actions Add Datafile Go
Select	Name ▲	Type	Extent Management	Segment Management	Status	Size (MB)	Used (MB)	Used (%)	
<input checked="" type="radio"/>	EXAMPLE	PERMANENT	LOCAL	AUTO	ONLINE	150.000	66.875	<div><div></div></div>	44.58
<input type="radio"/>	SYSAUX	PERMANENT	LOCAL	AUTO	ONLINE	230.000	222.688	<div><div></div></div>	96.82
<input type="radio"/>	SYSTEM	PERMANENT	LOCAL	MANUAL	ONLINE	440.000	434.375	<div><div></div></div>	98.72
<input type="radio"/>	TEMP	TEMPORARY	LOCAL	MANUAL	ONLINE	26.000	25.000	<div><div></div></div>	96.15
<input type="radio"/>	UNDOTBS1	UNDO	LOCAL	MANUAL	ONLINE	25.000	11.750	<div><div></div></div>	47.00
<input type="radio"/>	USERS	PERMANENT	LOCAL	AUTO	ONLINE	5.000	2.750	<div><div></div></div>	55.00



# Altering a Tablespace



## Edit Tablespace: EXAMPLE

Show SQL Revert Apply

**General** Storage Thresholds

Name

Bigfile tablespace **No**

**Extent Management**

- ☒ Locally Managed
- ☐ Dictionary Managed

**Type**

- ☒ Permanent
  - ☐ Set as default permanent tablespace
- ☐ Temporary
  - ☐ Set as default temporary tablespace
- ☐ Undo

**Status**

- ☒ Read Write
- ☐ Read Only
- ☐ Offline
  - Offline Mode 

Normal

Normal

Temporary

Immediate

For Recover

**Datafiles**

Add

Edit Remove

Select	Name	Directory	Size (MB)	Used (MB)
<input checked="" type="checkbox"/>	example01.dbf	/u01/app/oracle/product/10.1.0/oradata/orcl/	150.00	<div></div> 66.88



# Actions with Tablespaces

Delete Actions Generate DDL Create

Go

Used (MB)	Use	
66.875		Generate DDL
202.938		Make Locally Managed
427.125		Make Readonly
25.000		Make Writable
23.125		Place Online
2.750		Reorganize
		Show Dependencies
		Run Segment Advisor
		Take Offline

Edit Tablespace: EXAMPLE

Logged in As SYS

DDL


Return

```
CREATE SMALLFILE TABLESPACE "EXAMPLE" DATAFILE
'/u01/app/oracle/oradata/orcl/example01.dbf' SIZE 150M
REUSE AUTOEXTEND ON NEXT 640K MAXSIZE 32767M NOLOGGING
EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
BEGIN DBMS_SERVER_ALERT.SET_THRESHOLD
(9000, NULL, NULL, NULL, NULL, 1, 1, NULL, 5, 'EXAMPLE'); END;
```



# Dropping Tablespaces

Database: [orcl.us.oracle.com](#) > [Tablespaces](#) > Delete Tablespace: EXAMPLE [Logged in As SYS](#)

 **Warning**

[No](#) [Yes](#)


Once a tablespace has been dropped, the objects and data in it will no longer be available. To recover them can be a time consuming process. Oracle recommends a backup before and after dropping a tablespace.

Are you sure you want to delete Tablespace EXAMPLE?

☒ Delete associated datafiles from the OS

[No](#) [Yes](#)

[Edit](#) [View](#) [Delete](#) [Actions](#) [Run Segment Advisor](#) [Go](#)

Select	Name 	Type	Extent Management	Segment Management	Status	Size (MB)	Used (MB)	Used (%)
<input checked="" type="radio"/>	<a href="#">EXAMPLE</a>	PERMANENT	LOCAL	AUTO	ONLINE	150.000	66.875	<div><div></div></div> 44.58
<input type="radio"/>	<a href="#">SYSAUX</a>	PERMANENT	LOCAL	AUTO	ONLINE	220.000	211.313	<div><div></div></div> 96.05
<input type="radio"/>	<a href="#">SYSTEM</a>	PERMANENT	LOCAL	MANUAL	ONLINE	430.000	427.313	<div><div></div></div> 99.38
<input type="radio"/>	<a href="#">TEMP</a>	TEMPORARY	LOCAL	MANUAL	ONLINE	26.000	25.000	<div><div></div></div> 96.15
<input type="radio"/>	<a href="#">UNDOTBS1</a>	UNDO	LOCAL	MANUAL	ONLINE	25.000	11.938	<div><div></div></div> 47.75
<input type="radio"/>	<a href="#">USERS</a>	PERMANENT	LOCAL	AUTO	ONLINE	5.000	2.750	<div><div></div></div> 55.00



# Viewing Tablespace Information

Database: [orcl.oracle.com](#) > [Tablespaces](#) > View Tablespace: EXAMPLE Logged in As SYS

## View Tablespace: EXAMPLE

[Edit](#) [Return](#)

Name **EXAMPLE**  
Bigfile tablespace **No**  
Status **ReadWrite**  
Type **Permanent**  
Extent Management **local**

### Storage

Allocation Type **Automatic**  
Segment Space Management **Automatic**  
Enable logging **No**  
Block Size (B) **8192**

### Datafiles

Name	Directory	Size (MB)	Used (MB)
<a href="#">example01.dbf</a>	/u01/app/oracle/oradata/orcl/	150.00	<div><div></div></div> 80.25

### Thresholds

Use Default Thresholds  
Warning (% used) **85**  
Critical (% used)

[Edit](#) [View](#) [Delete](#) Actions [Get](#)

Select	Name	Type	Extent Management	Segment Management	Status	Size (MB)	Used (MB)	Use
<input checked="" type="radio"/>	EXAMPLE	PERMANENT	LOCAL	AUTO	ONLINE	150.000	80.250	<div><div></div></div>



# Locally Managed Tablespaces

- Fast, concurrent space operations. Space allocations and deallocations modify locally managed resources (bitmaps stored in header files).
- Enhanced performance
- Readable standby databases are allowed, because locally managed temporary tablespaces do not generate any undo or redo.
- Space allocation is simplified, because when the AUTOALLOCATE clause is specified, the database automatically selects the appropriate extent size.
- User reliance on the data dictionary is reduced, because the necessary information is stored in file headers and bitmap blocks.
- Coalescing free extents is unnecessary for locally managed tablespaces.



# Bigfile Tablespaces

- A **bigfile tablespace** is a tablespace with a single, but very large (up to 4G blocks) datafile. Traditional smallfile tablespaces, in contrast, can contain multiple datafiles, but the files cannot be as large.
- Bigfile tablespaces can reduce the number of datafiles needed for a database.



# Temporary Tablespaces

- A temporary tablespace contains transient data that persists only for the duration of the session.
- It improve the concurrency of multiple sort operations, reduce their overhead, and avoid Oracle Database space management operations



# Altering Tablespace Availability

- Taking Tablespaces Offline
- Bringing Tablespaces Online





## Using Read-Only Tablespaces

- Making a tablespace read-only prevents write operations on the datafiles in the tablespace.
- The primary purpose of read-only tablespaces is to eliminate the need to perform backup and recovery of large, static portions of a database.
- Read-only tablespaces also provide a way to protecting historical data so that users cannot modify it.
- Making a tablespace read-only prevents updates on all tables in the tablespace, regardless of a user's update privilege level.



# Renaming Tablespaces

- Using the RENAME TO clause of the ALTER TABLESPACE, you can rename a permanent or temporary tablespace.
- For example, the following statement renames the users tablespace:
- ALTER TABLESPACE users RENAME TO usersts;



# Lesson Objectives

After completing this lesson, you should be able to do the following:

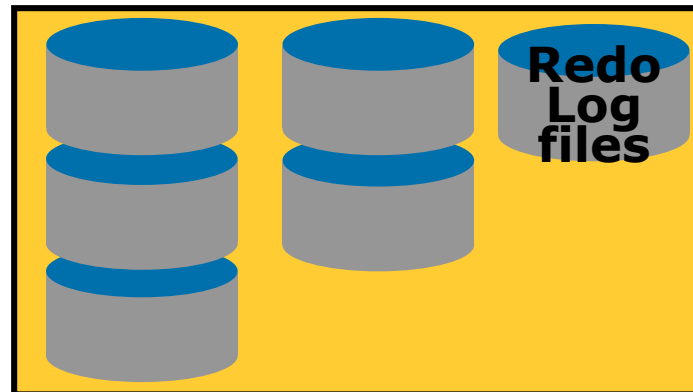
- Explain the purpose of online redo log files
- Outline the structure of online redo log files
- Control log switches and checkpoints
- Multiplex and maintain online redo log files
- Manage online redo logs files with OMF
- What Is the Archived Redo Log?
- Choosing Between NOARCHIVELOG and ARCHIVELOG Mode



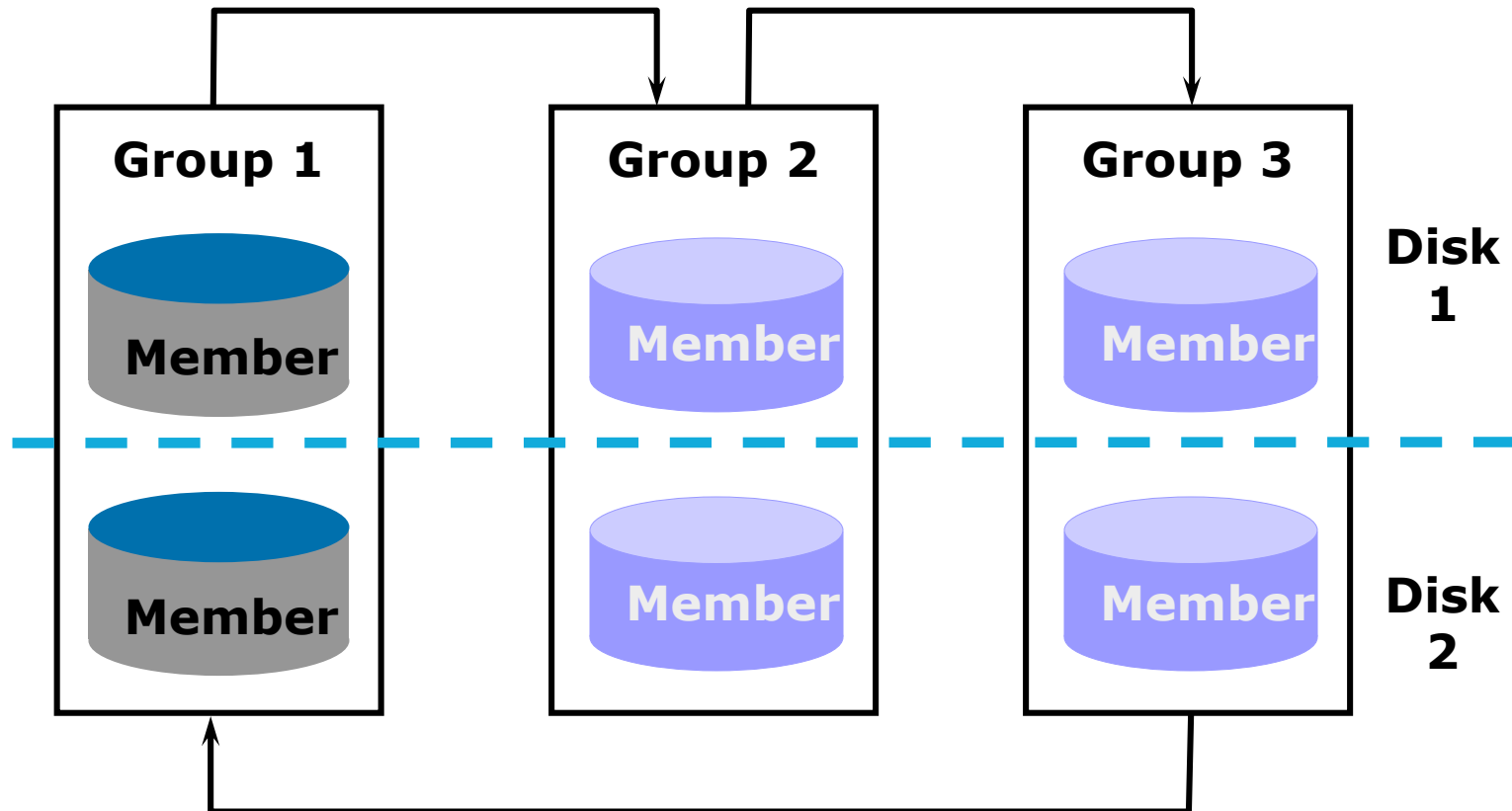


# Using Redo Log Files

- Redo log files have the following characteristics:
  - Record all changes made to data
  - Provide a recovery mechanism
  - Can be organized into groups
  - At least two groups required



# Structure of Redo Log Files





## How Redo Log Files Work

- Redo log files are used in a cyclic fashion.
- When a redo log file is full, LGWR will move to the next log group.
  - Called a log switch
  - Checkpoint operation also occurs
  - Information written to the control file



# Forcing Log Switches and Checkpoints

- Forcing a log switch:

```
ALTER SYSTEM SWITCH LOGFILE;
```

- Checkpoints can be forced by using:
  - Setting FAST\_START\_MTTR\_TARGET parameter

```
FAST_START_MTTR_TARGET = 600
```

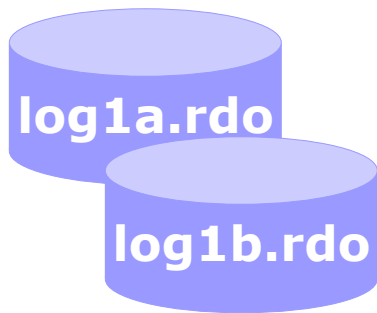
- ALTER SYSTEM CHECKPOINT command

```
ALTER SYSTEM CHECKPOINT;
```

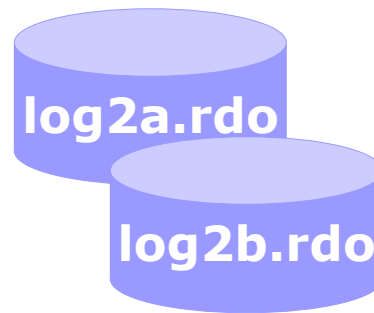


## Adding Online Redo Log File Groups

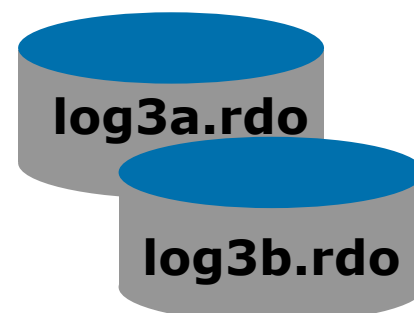
```
ALTER DATABASE ADD LOGFILE GROUP 3  
('$HOME/ORADATA/u01/log3a.rdo',  
'$HOME/ORADATA/u02/log3b.rdo')  
SIZE 1M;
```



**Group  
1**



**Group  
2**



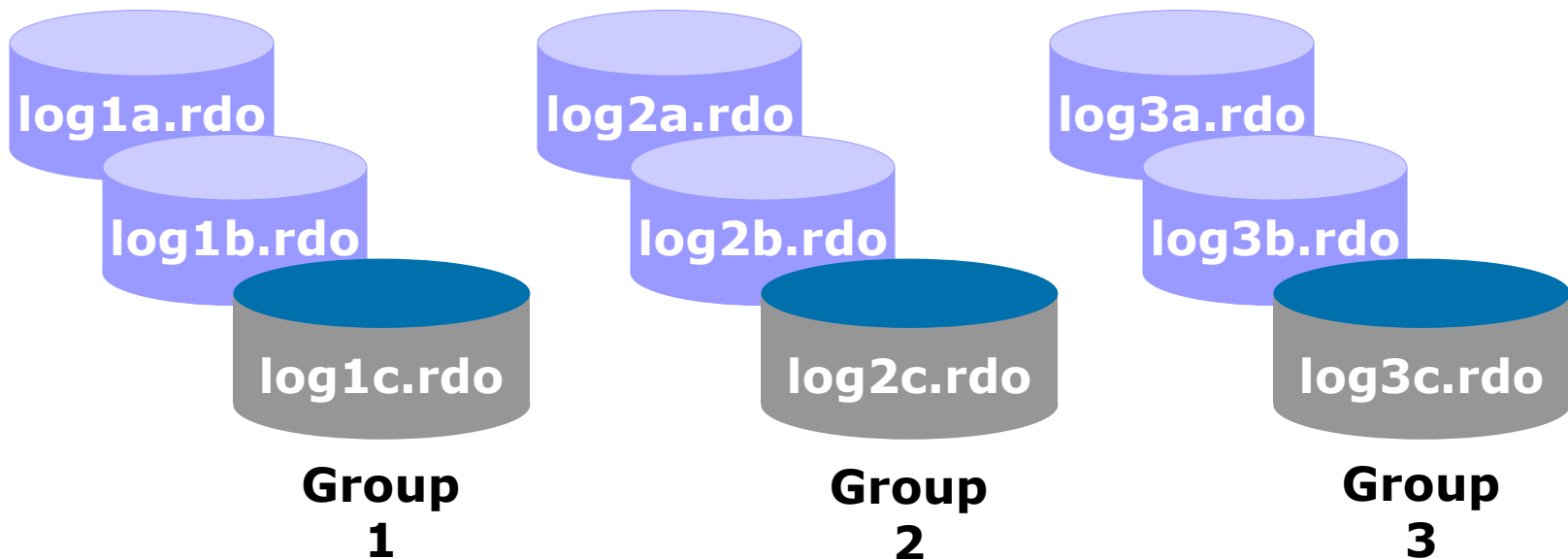
**Group  
3**





## Adding Online Redo Log File Members

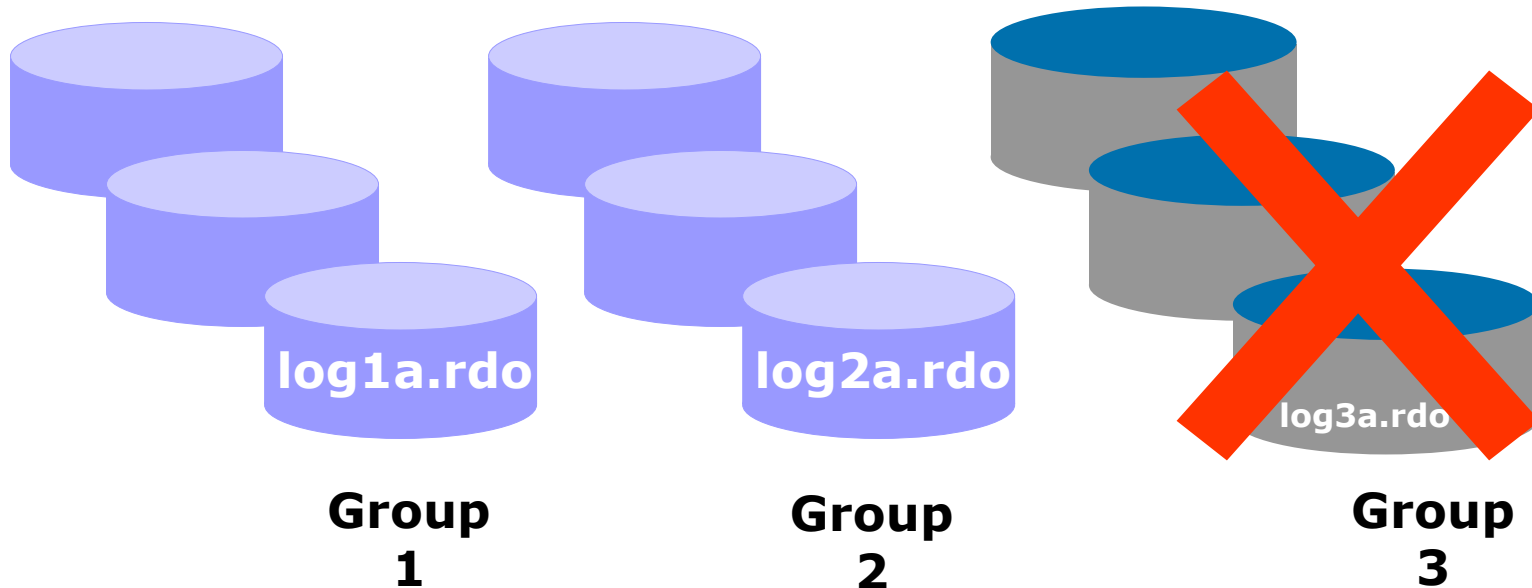
```
ALTER DATABASE ADD LOGFILE MEMBER  
'$HOME/ORADATA/u04/log1c.rdo' TO GROUP 1,  
'$HOME/ORADATA/u04/log2c.rdo' TO GROUP 2,  
'$HOME/ORADATA/u04/log3c.rdo' TO GROUP 3;
```





# Dropping Online Redo Log File Groups

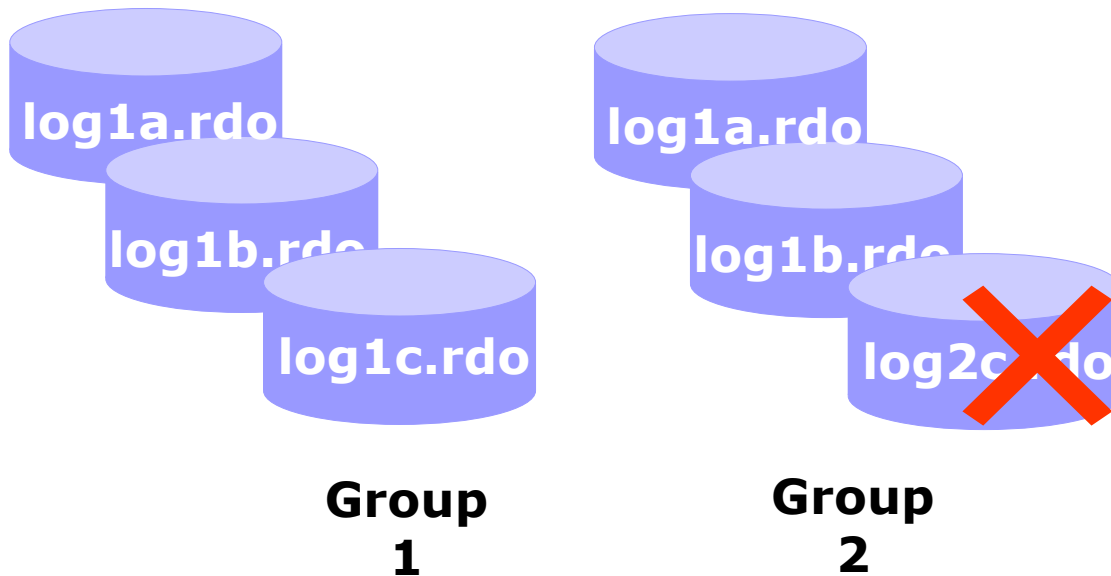
```
ALTER DATABASE DROP LOGFILE GROUP 3;
```





# Dropping Online Redo Log File Members

```
ALTER DATABASE DROP LOGFILE MEMBER  
'$HOME/ORADATA/u04/log3c.rdo';
```





## Relocating or Renaming Online Redo Log Files

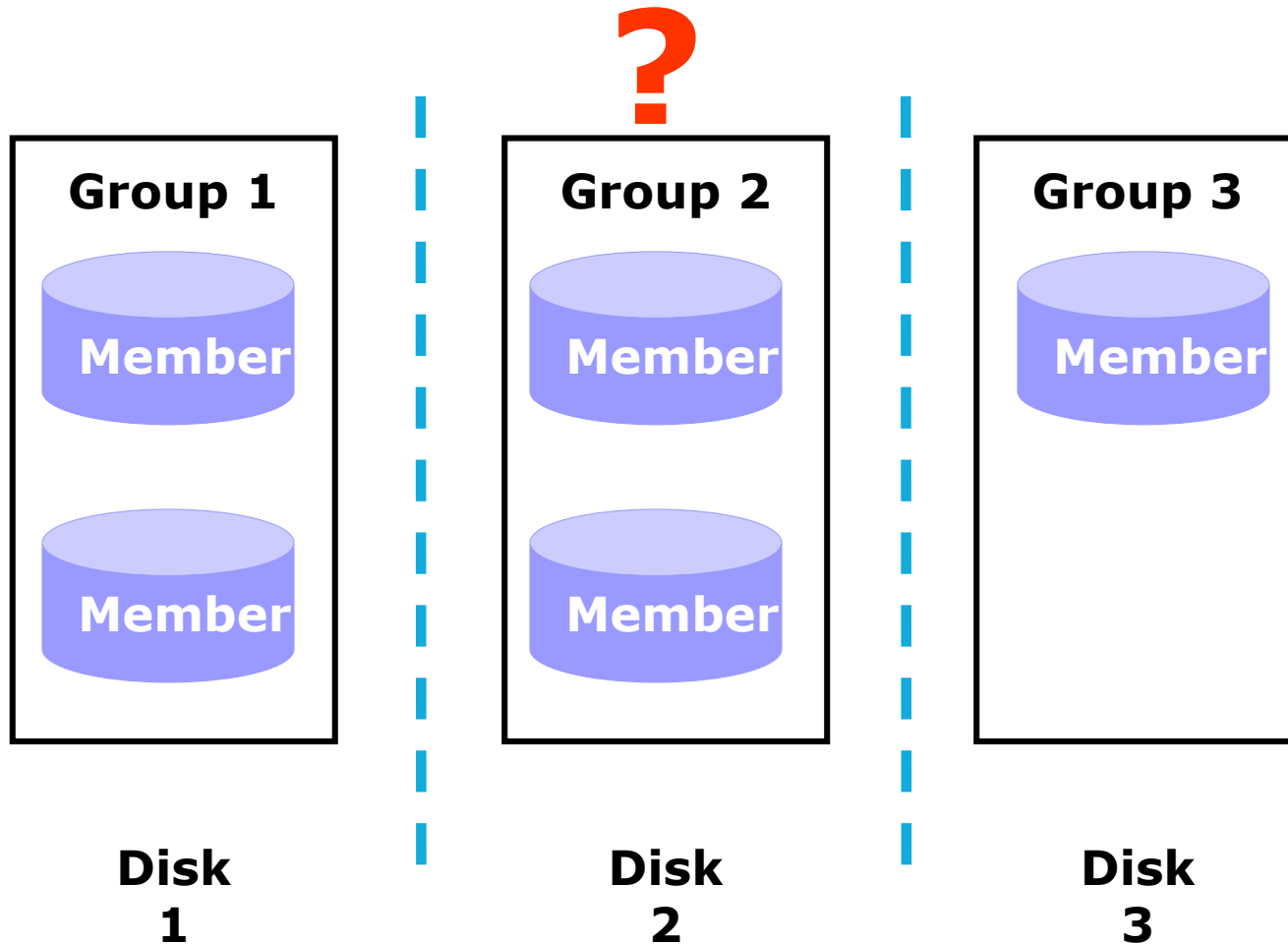
Relocate or rename online redo log files in one of the two following ways:

- ALTER DATABASE CLEAR LOGFILE command
  - Copy the online redo log files to the new location
  - Execute the command

```
ALTER DATABASE CLEAR LOGFILE  
'$HOME/ORADATA/u01/log2a.rdo';
```

- Add new members and drop old members

# Online Redo Log File Configuration





## Managing Online Redo Log Files with OMF

- Define the DB\_CREATE\_ONLINE\_LOG\_DEST\_n parameter:

```
DB_CREATE_ONLINE_LOG_DEST_1  
DB_CREATE_ONLINE_LOG_DEST_2
```

- Group can be added with no file specification:

```
ALTER DATABASE ADD LOGFILE;
```

- Dropping a group:

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



# Obtaining Group and Member Information

Information about a group and its members can be obtained by querying the following views:

- V\$LOG
- V\$LOGFILE



# What Is the Archived Redo Log?

Oracle Database lets you save filled groups of redo log files to one or more offline destinations, known collectively as the archived redo log, or more simply the archive log.

The process of turning redo log files into archived redo log files is called archiving.

This process is only possible if the database is running in ARCHIVELOG mode.

You can choose automatic or manual archiving.





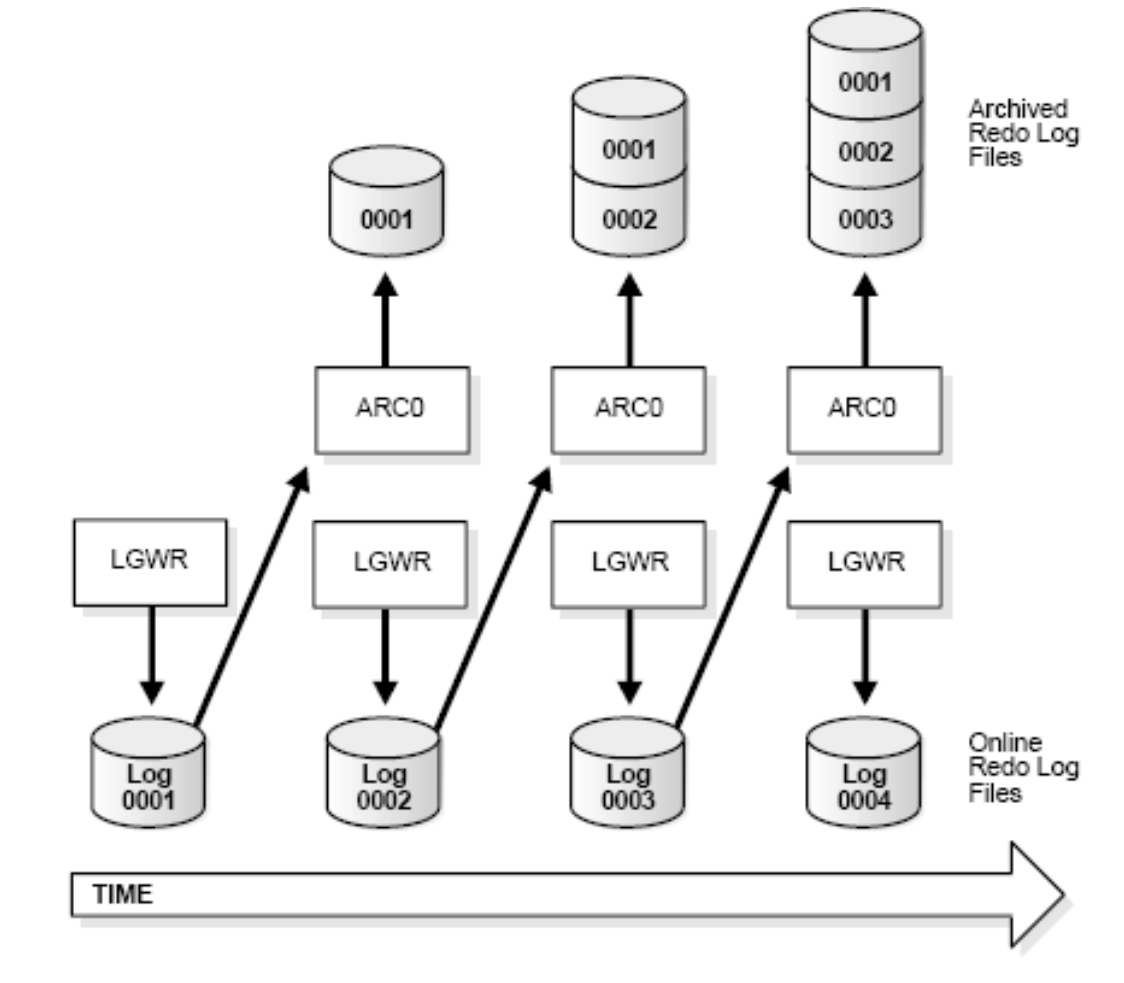
# Choosing Between NOARCHIVELOG and ARCHIVELOG Mode

## Running a Database in NOARCHIVELOG Mode

- When you run your database in NOARCHIVELOG mode, you disable the archiving of the redo log.
- The database control file indicates that filled groups are not required to be archived.
- Therefore, when a filled group becomes inactive after a log switch, the group is available for reuse by LGWR.
- When you run a database in ARCHIVELOG mode, you enable the archiving of the redo log.
- The database control file indicates that a group of filled redo log files cannot be reused by LGWR until the group is archived.
- A filled group becomes available for archiving immediately after a redo log switch occurs.



## Redo Log File Use in ARCHIVELOG Mode





## Contd...Archived Redo Log Files

- Filled online redo log files can be archived.
- There are two advantages in running the database in ARCHIVELOG mode and archiving redo log files:
  - Recovery: A database backup together with online and archived redo log files can guarantee recovery of all committed transactions.
  - Backup: This can be performed while the database is open.
- By default, database is created in NOARCHIVELOG mode.



## Contd...Archived Redo Log Files

- Accomplished automatically by ARCn
- Accomplished manually through SQL statements
- When successfully archived:
  - An entry in the control file is made
  - Records: archive log name, log sequence number, and high and low system change number (SCN)
  - Filled redo log file cannot be reused until:
    - A checkpoint has taken place
    - File has been archived by ARCn
- Can be multiplexed
- Maintained by the DBA

# SUMMARY

- In this lesson, you should have learned how to:
  - Explain the use of online redo log files
  - Obtain redo log file information
  - Control log switches and checkpoints
  - Multiplex and maintain online redo log files
  - Manage online redo log files with OMF
  - What Is the Archived Redo Log?
  - Choosing Between NOARCHIVELOG and ARCHIVELOG Mode