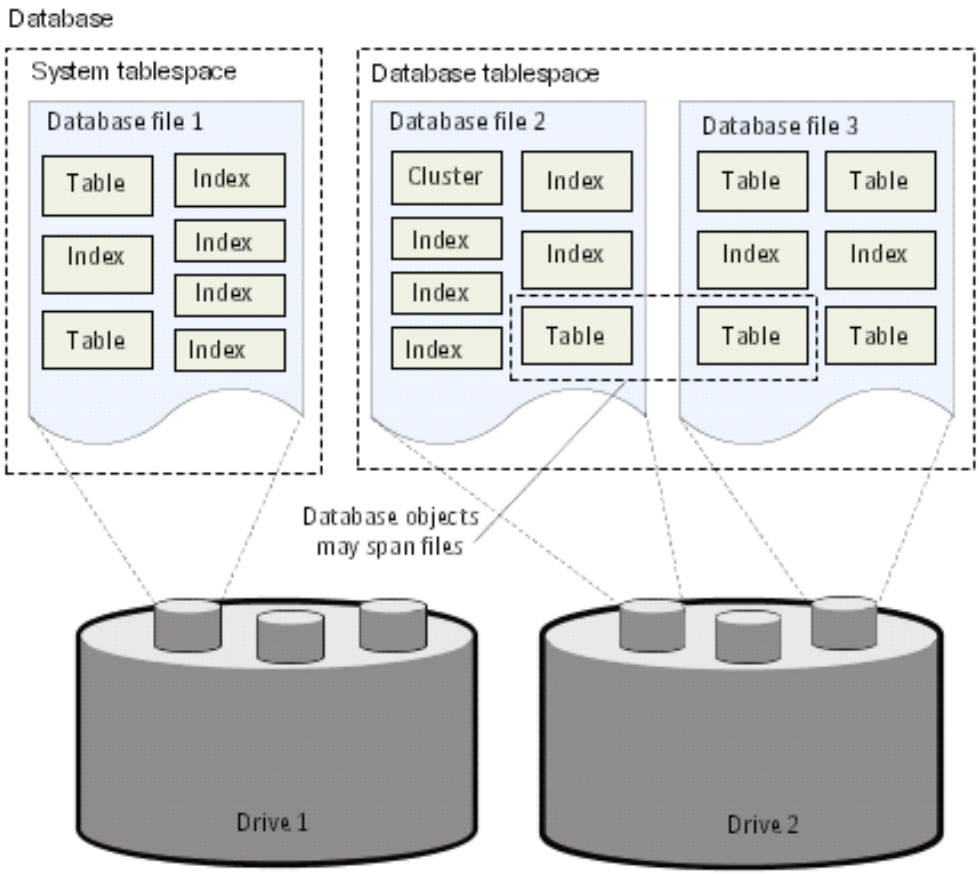


数据库引航 第七课—Oracle数据库对象（二）

2012.10.15

重做日志 --REDO

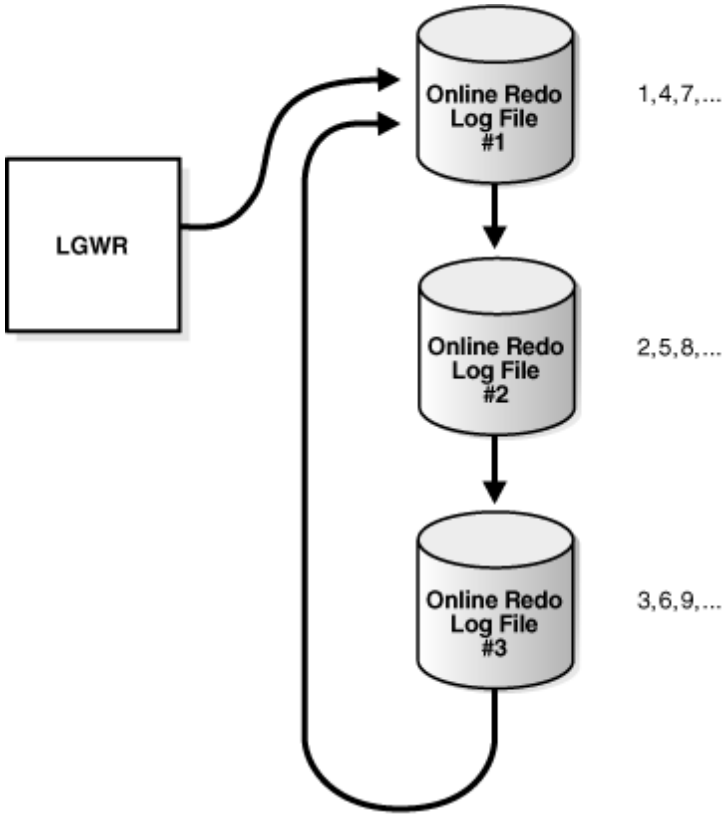
什么是重做?

- 重做日志 包含所有数据产生的历史改变记录
- 重做日志文件 通常用于
  - 恢复(实例恢复和介质恢复)
  - 日志挖掘
  - 流

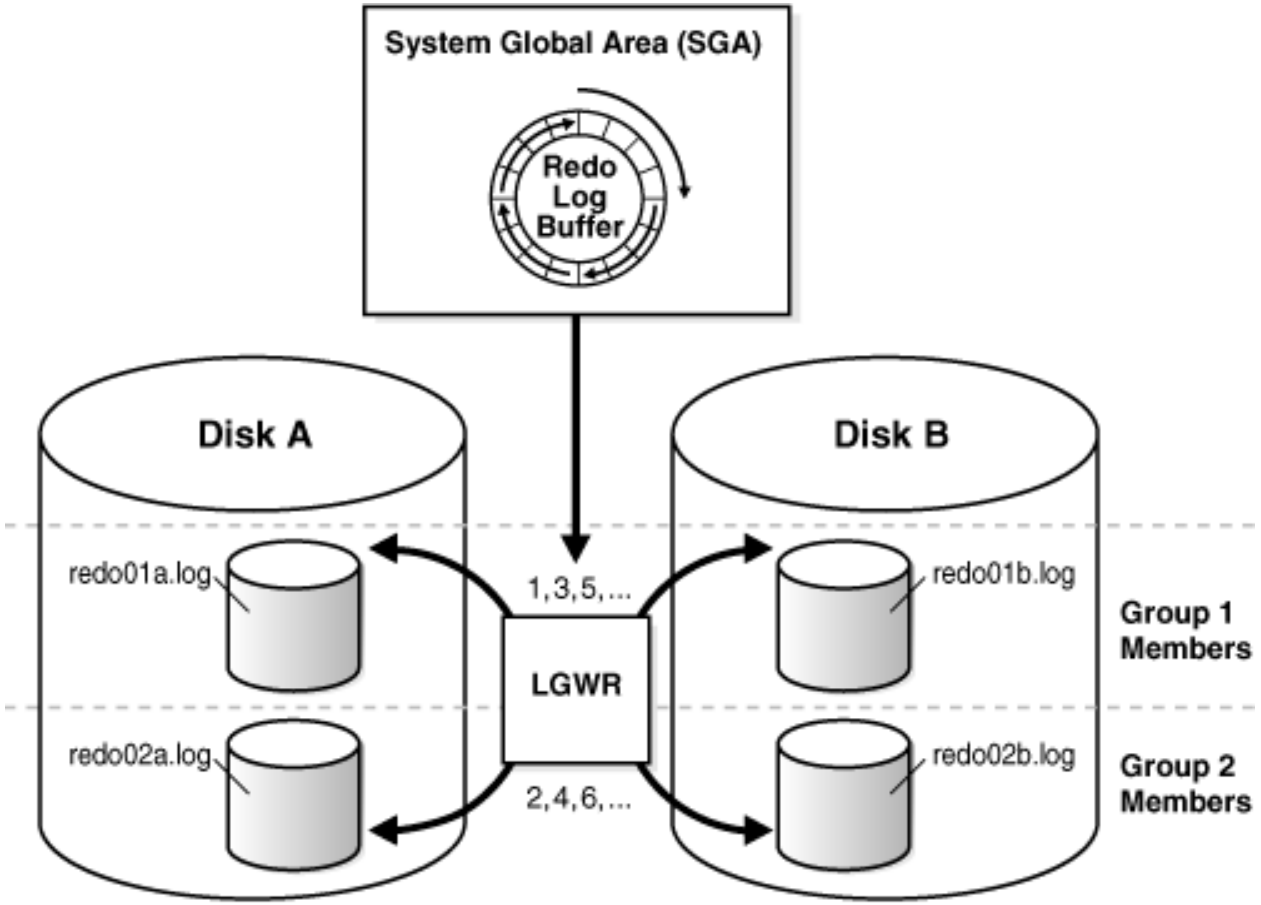
  - 数据库产生的每个改动：
  - 写入数据块缓冲之前，先写入重做日志 －－内存
  - 写入数据文件之前先写入日志文件 －－数据文件

- 当提交后，日志缓冲被刷入重做文件里。

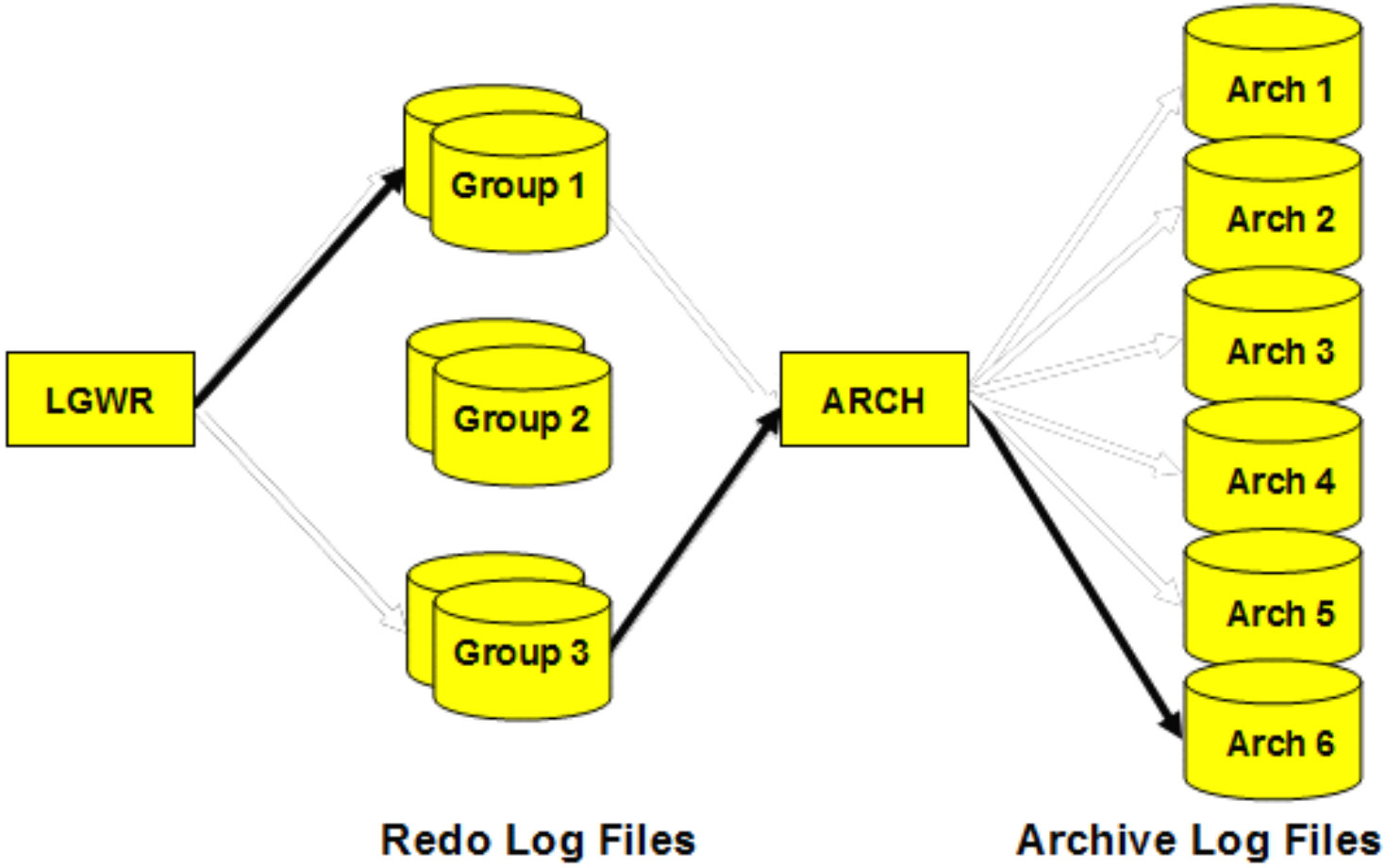
REDO


LGWR
Online Redo Log File #1
Online Redo Log File #2
Online Redo Log File #3
1,4,7,...
2,5,8,...
3,6,9,...

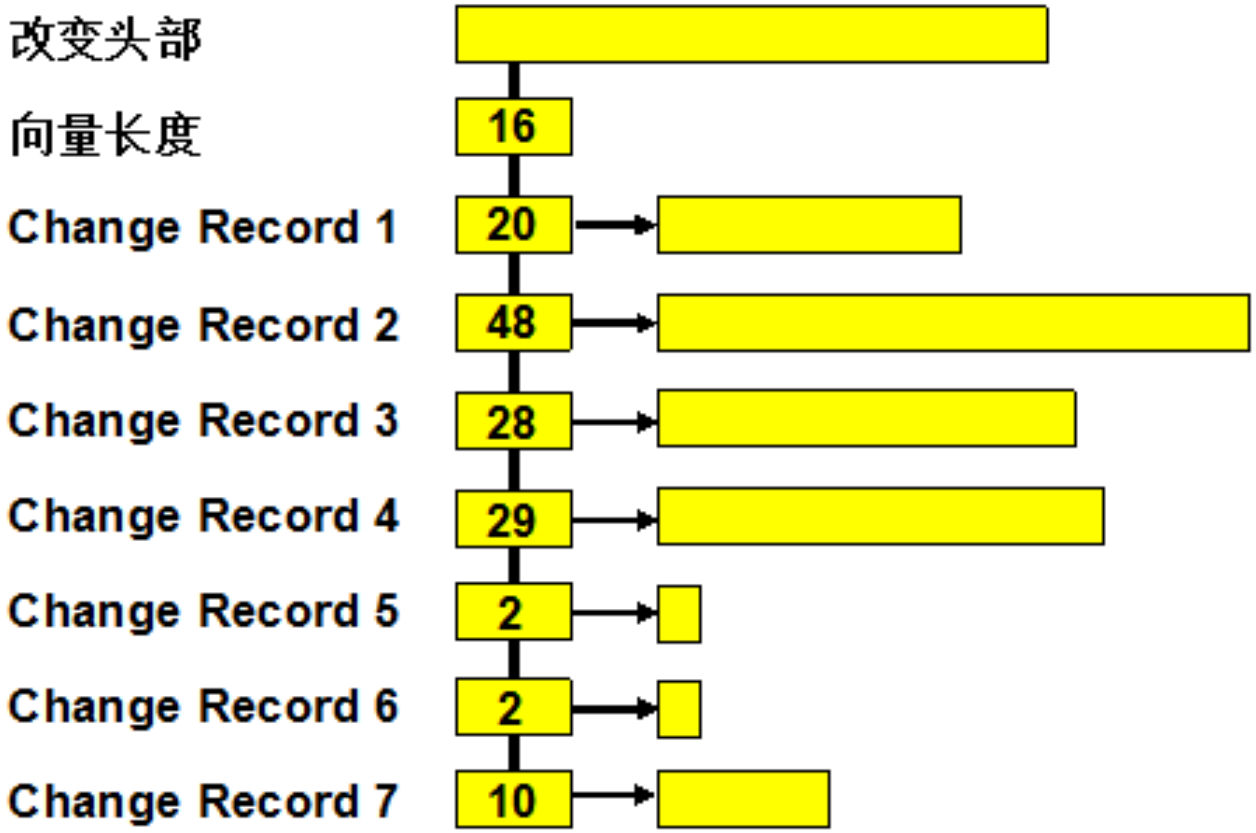
REDO--日志组


System Global Area (SGA)
Redo Log Buffer
Disk A
Disk B
redo01a.log
redo02a.log
redo01b.log
redo02b.log
1,3,5,...
2,4,6,...
LGWR
Group 1 Members
Group 2 Members

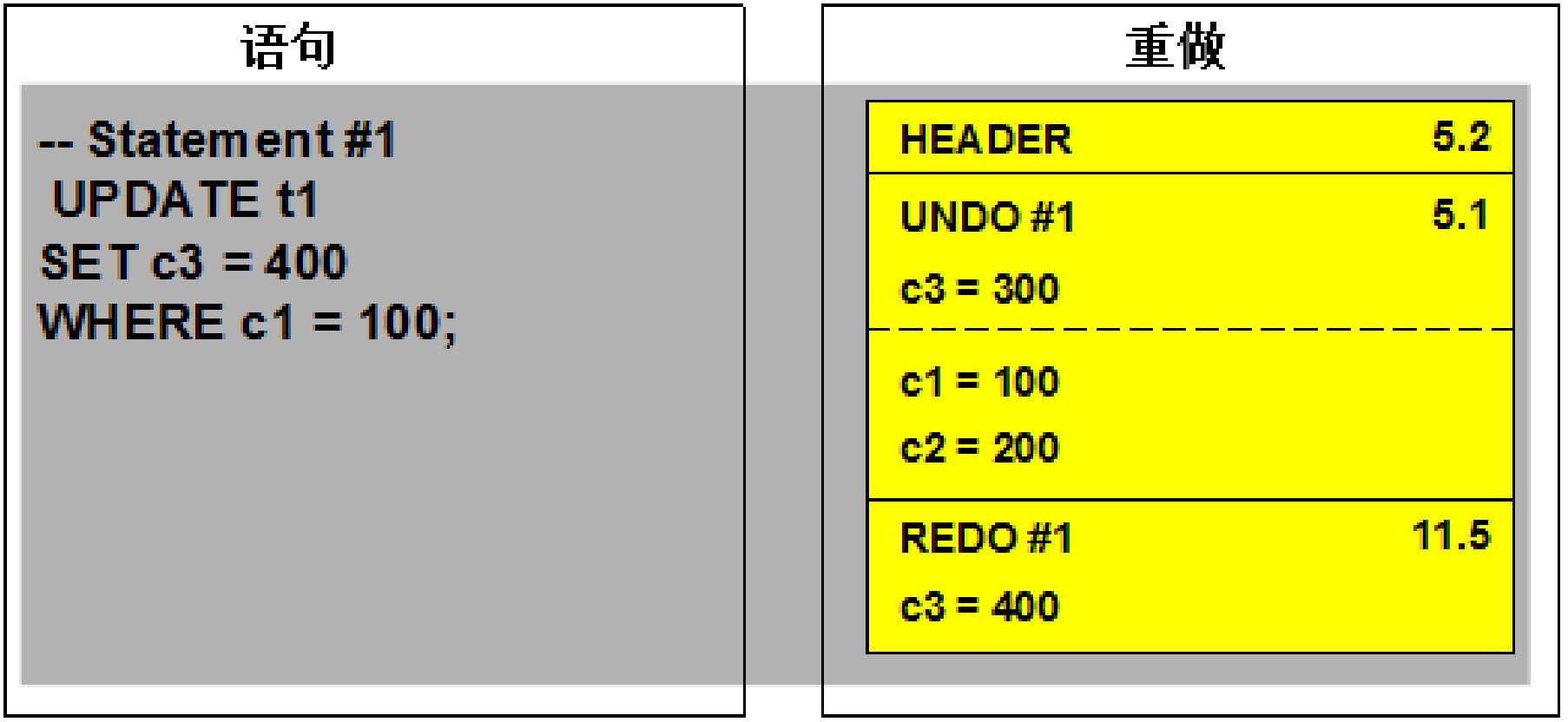
REDO和归档


LGWR
Group 1
Group 2
Group 3
ARCH
Arch 1
Arch 2
Arch 3
Arch 4
Arch 5
Arch 6
Redo Log Files
Archive Log Files

REDO的内容---改变向量



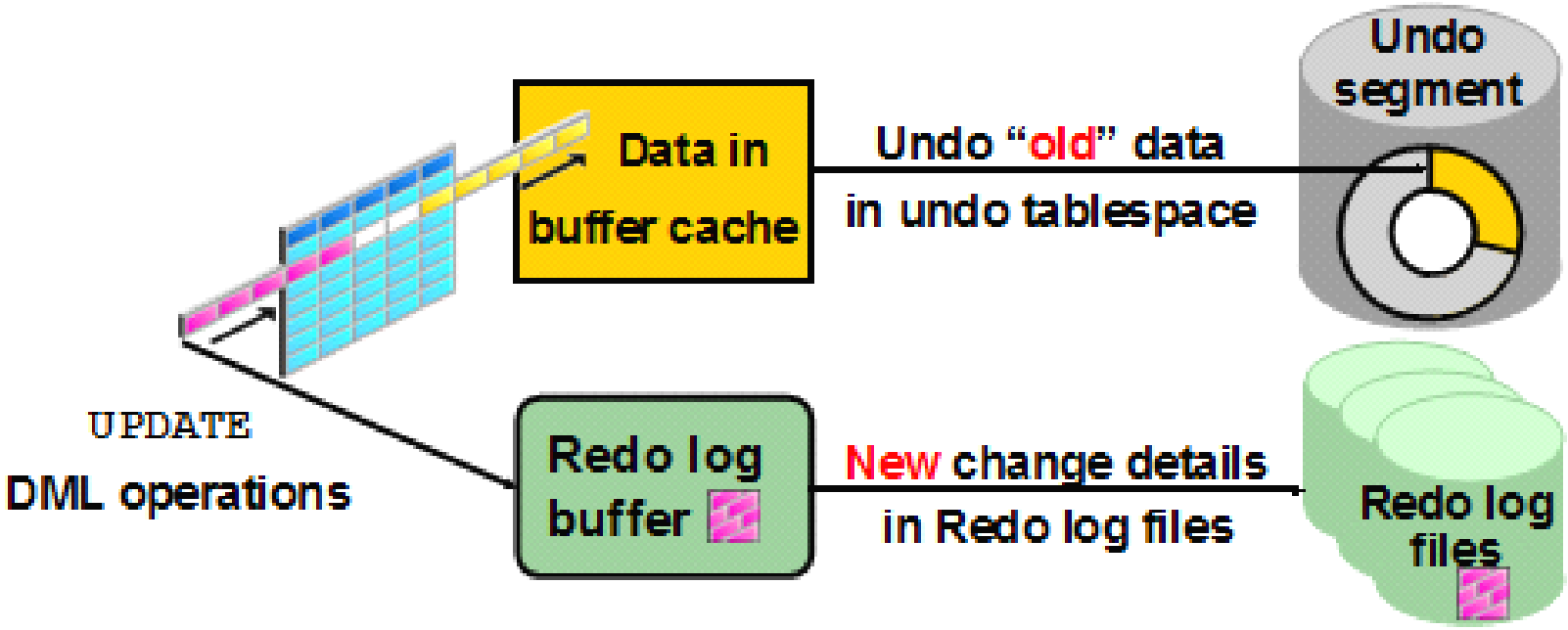
REDO --undo的redo



UNDO

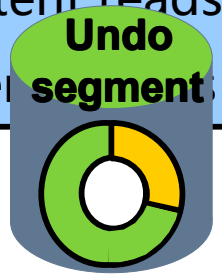
- 作用
  - 数据的回滚
  - 一致性读
  - 表的闪回（事务，查询的闪回...）
  - 失败会话的恢复

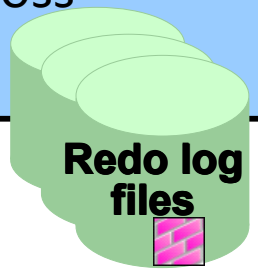
Undo的产生


Undo segment
Data in buffer cache
Undo "old" data in undo tablespace
UPDATE
DML operations
Redo log buffer
New change details in Redo log files
Redo log files

Undo V.S. Redo

	Undo	Redo
Record of	How to undo a change	How to reproduce a change
Used for	Rollback, read consistency, flashback	Rolling forward database changes
Stored in	Undo segments	Redo log files
Protects against	Inconsistent reads in multiuser	Data loss



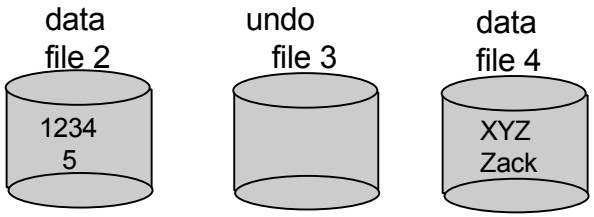


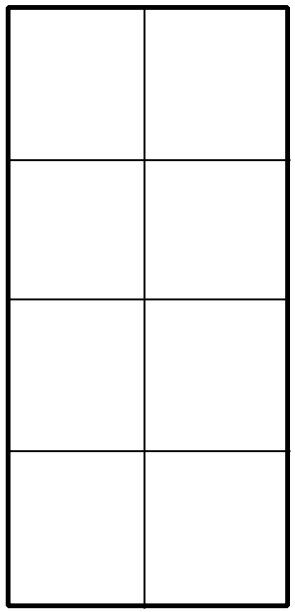
Redo Buffer Operation

When user enters…

User #1

```
UPDATE emp
SET sal = 10
WHERE id = 1234
```





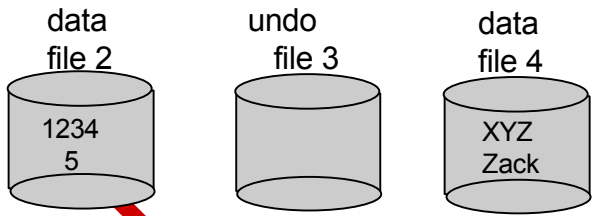
database buffer cache

Redo Buffer Operation

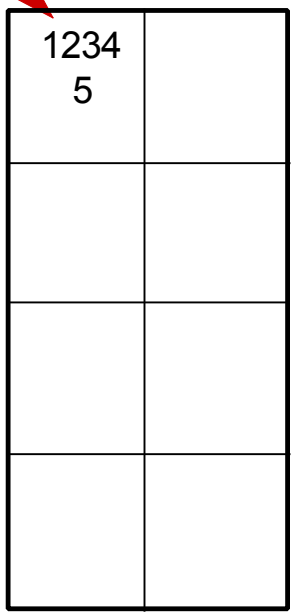
When user enters…

User #1

UPDATE emp
SET sal = 10
WHERE id = 1234



The **data** block is retrieved from disk to the database buffer cache (unless it's already there)



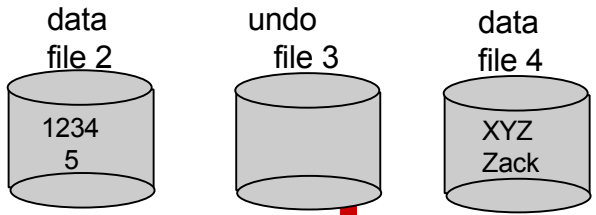
database buffer cache

Redo Buffer Operation

When user enters…

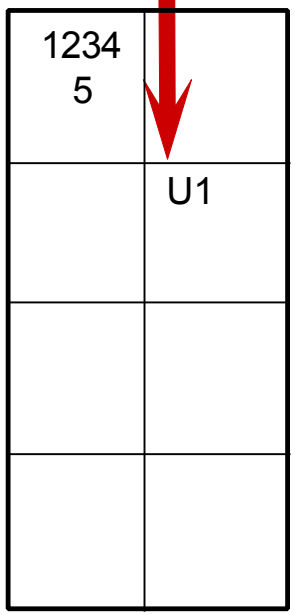
User #1

UPDATE emp
SET sal = 10
WHERE id = 1234



The **data** block is retrieved from disk to the database buffer cache (unless it's already there)

The **undo** block is then retrieved from disk to the database buffer cache (unless it's already there)



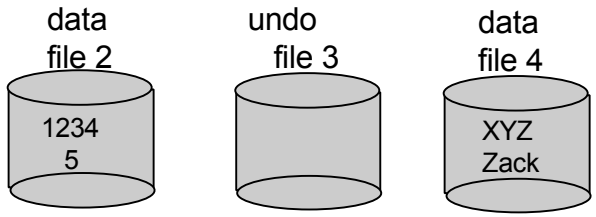
database buffer cache

Redo Buffer Operation

When user enters…

User #1

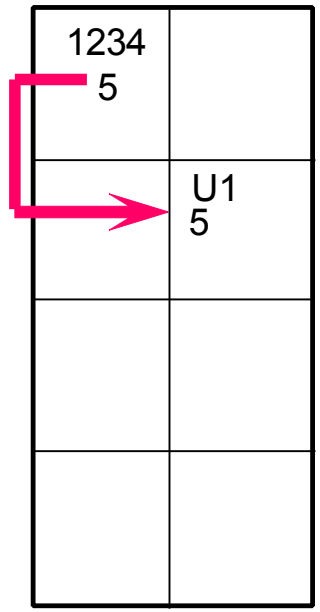
UPDATE emp
SET sal = 10
WHERE id = 1234



The **data** block is retrieved from disk to the database buffer cache (unless it's already there)

The **undo** block is then retrieved from disk to the database buffer cache (unless it's already there)

The **undo** data is created showing the 'before image'



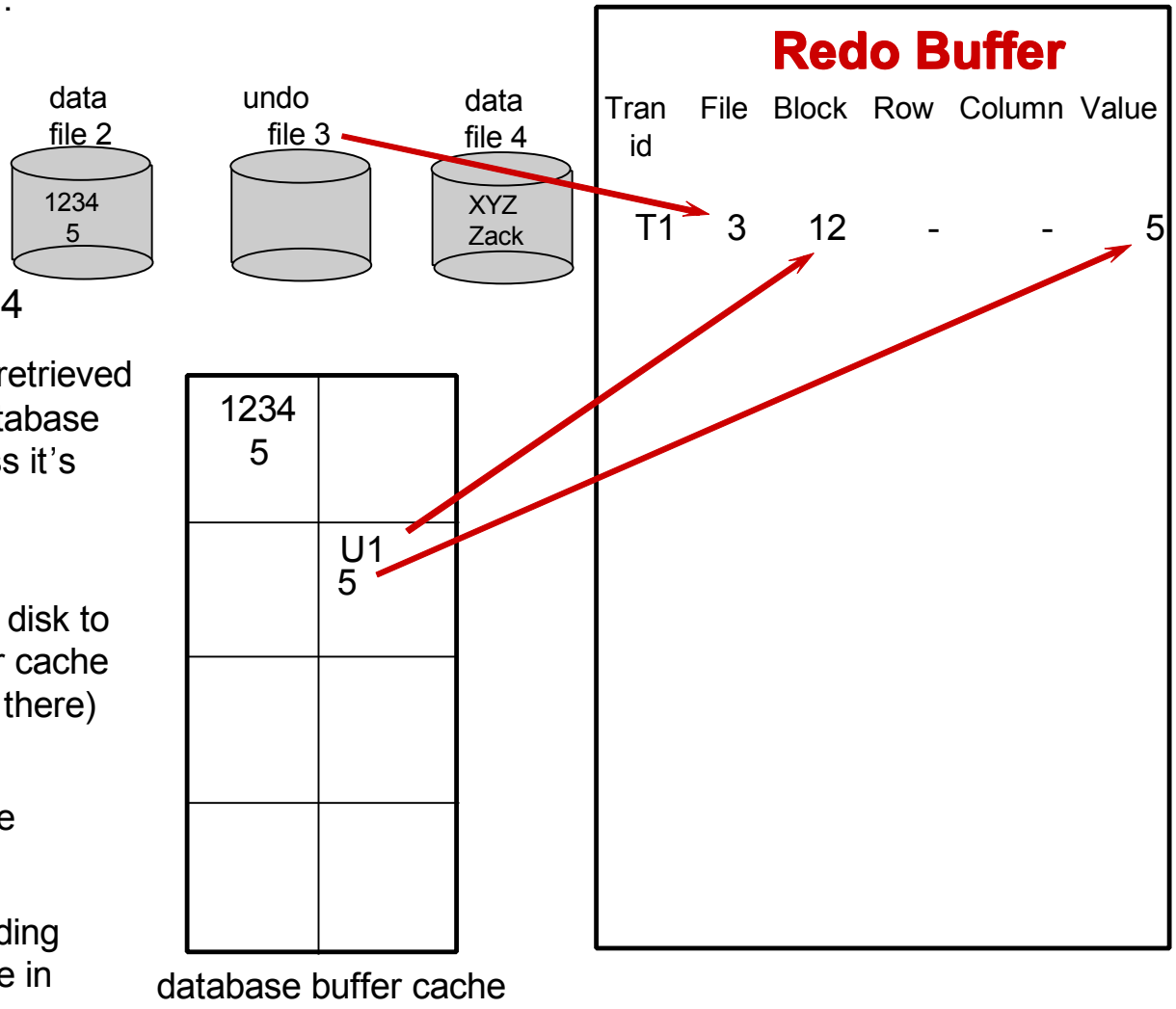
database buffer cache

Redo Buffer Operation

When user enters…

User #1

UPDATE emp
SET sal = 10
WHERE id = 1234



The **data** block is retrieved from disk to the database buffer cache (unless it's already there)

The **udo** block is then retrieved from disk to the database buffer cache (unless it's already there)

The **undo** data is created showing the 'before image'

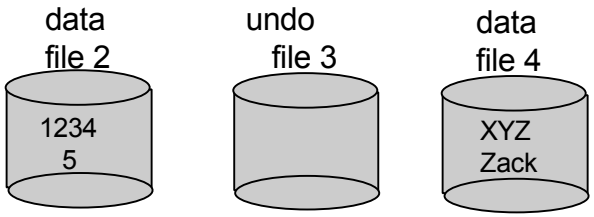
And an entry recording this change is made in the **REDO** buffer

Redo Buffer Operation

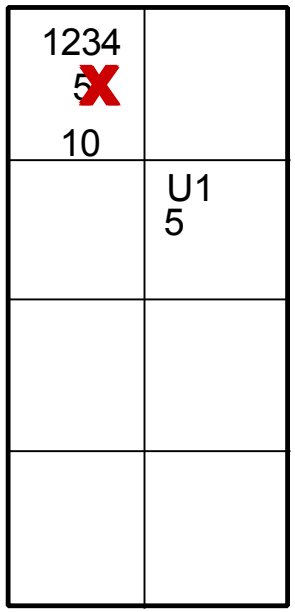
When user enters…

User #1

UPDATE emp
SET sal = 10
WHERE id = 1234



Then the data block is changed...



database buffer cache

Redo Buffer

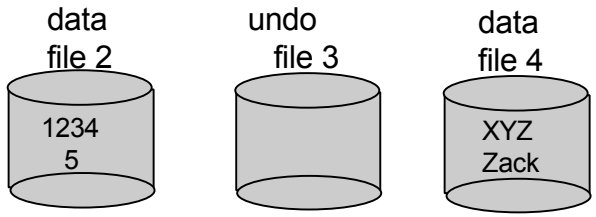
Tran id	File	Block	Row	Column	Value
T1	3	12	-	-	5

Redo Buffer Operation

When user enters…

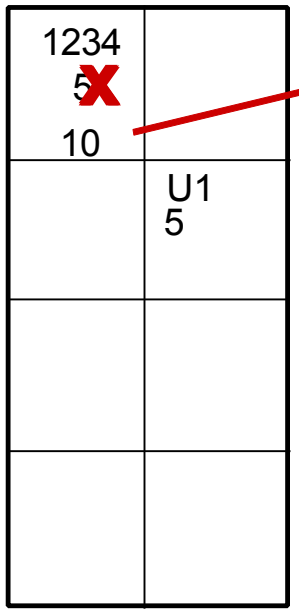
User #1

UPDATE emp
SET sal = 10
WHERE id = 1234



Then the data block is changed...

And an entry recording this change is made in the **REDO** buffer



database buffer cache

Redo Buffer

Tran id	File	Block	Row	Column	Value
T1	3	12	-	-	5
T1	2	123	41	6	10

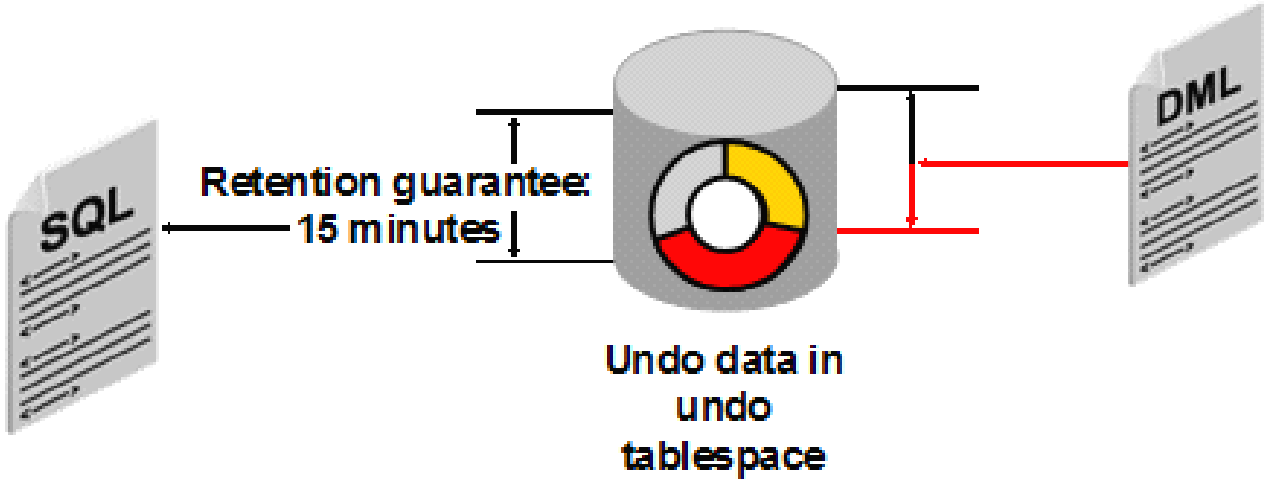
Undo 表空间

- 普通的数据表空间
- 机制和数据表空间完全一致，只不过用途不同。

自动管理回滚段表空间

- 动态改变回滚段的个数。
- 自动调整回滚段的大小

Undo数据的保留时间



SELECT statements
running 15 minutes or less
are always satisfied.

A transaction will fail
if it generates more undo
than there is space.

Undo表空间的损坏

- 如果有活动的事务，需要恢复Undo表空间，如果没有备份，将导致数据库损坏。
- 如果没有活动事务，可以重建一个undo表空间。

Thanks

FAQ时间