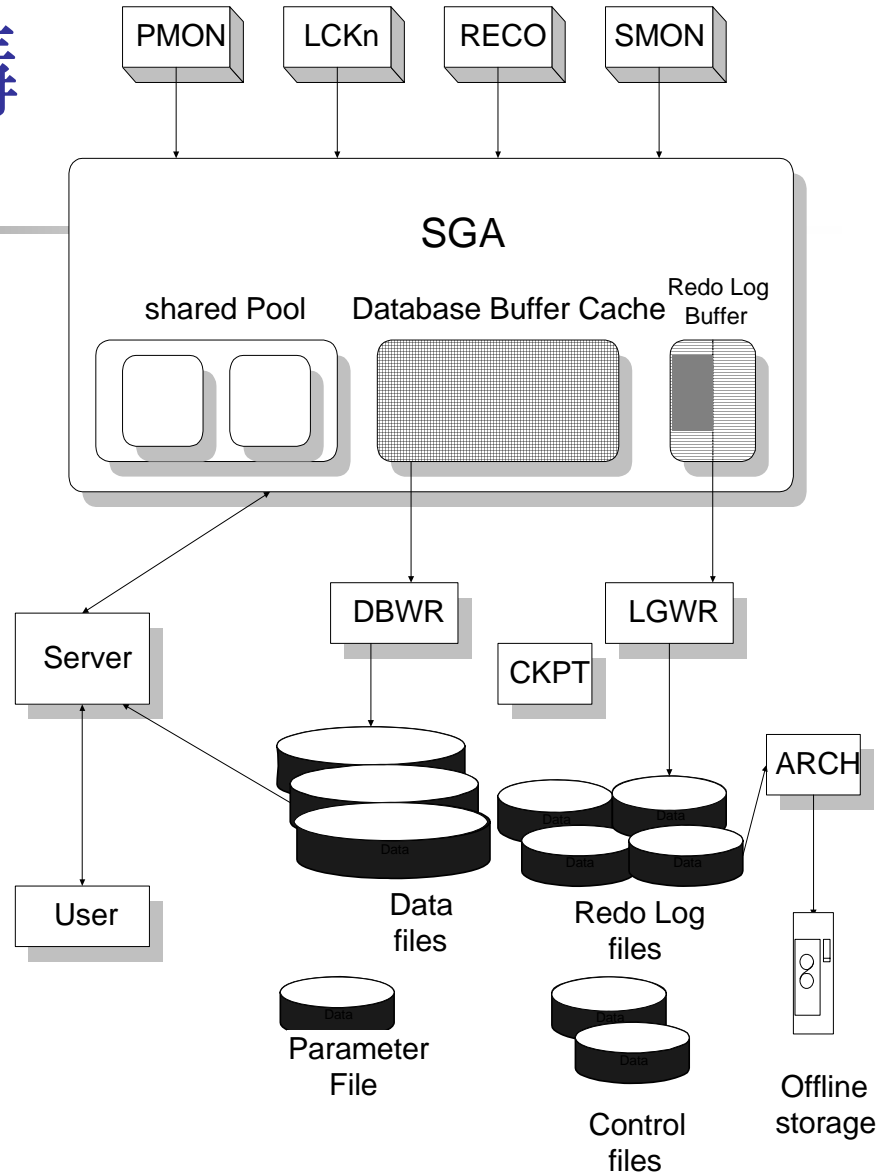




ORACLE 系統架構

Oracle 系統架構





Oracle 系統架構(Cont.)

- ❖ 整個 Oracle 系統架構包含以下三個部分，如圖一所示：
 1. System Global Area (系統共同區，SGA)
 2. Process (處理單元)
 3. Files (系統檔案)



System Global Area (SGA)

- ❖ 每當 Oracle 系統啓動時，會佔用主機一大塊資料庫專屬的記憶體空間來控制資訊與儲存資料，這塊資料庫專屬的記憶體空間稱爲 System Global Area(SGA)。

- 
-
- ❖ SGA 主要是由三個主要部份所組成：
 1. Shared
 2. Database Buffer Cache
 3. Redo Log Buffer



Shared Pool

- Shared Pool 主要是由 Library Cache、Dictionary Cache 與一些控制結構 (control structures) 所組成。



Database Buffer Cache

- 用來存放最近由資料庫讀出的資料。



Redo Log Buffer

- 爲了加快線上交易的效率，異動交易的記錄之內容並不直接寫入磁碟檔案中，而是存放在 Redo Log Buffer 這記憶體空間中。



Process (處理單元)

- ❖ 在 Oracle 中的 Process 分爲：
 1. User Process
 2. Oracle Process



User Process

當使用者的應用程式欲以 SQL 指令存取資料庫資料時，例如：Pro*C程式、Oracle Tools、SQL*plus、Oracle Form 等等，Oracle 會產生 User Process去執行這些工作。



Oracle Process

- Oracle Process 依執行的方式不同可大概分爲 Server Process 與 Background Process 。



Background Processes

- Database Writer (DBWR)
- Log Writer (LGWR)
- Process Monitor (PMON)
- System Monitor (SMON)
- Checkpoint (CKPT)
- Archiver (ARCH)
- Recoverer (RECO)
- LOCK (LCK_n)



Database Writer (DBWR)

當 Oracle 系統存取資料時，爲了考慮執行效率，並不會直接存取磁碟中的資料，而是先到 SGA 中的 Database Buffer Cache 中去找，若找不到再去存取磁碟中的資料。



Log Writer (LGWR)

Oracle 系統將使用者所執行的異動交易 (Transaction) 記錄在日誌檔 (Redo Log File) 中，以供 Oracle 系統回復資料之用。



Process Monitor (PMON)

PMON 監控所有 client 端的連線程式，並在 User Process 處理失敗時，負責回收使用者所佔用的系統資源。



System Monitor (SMON)

SMON 監控所有系統資料的正確性，當資料庫開啓時檢查資料的一致性，一旦發現了資料有不一致性時會進行 Crash Recovery 資料回復動作。



Checkpoint (CKPT)

每隔一段固定的時間 DBWR 會將 SGA 的 Database Buffer Cache 中已修過的資料寫回 Data File 中，這個時間點稱為 Checkpoint，而 CKPT 的任務是負責當 Checkpoint 時間到了時告訴 DBWR 要進行資料更新動作。



Archiver (ARCH)

在 LGWR 重頭開始寫入第一個 Redo Log File 之前， ARCH 負責將 Redo Log File 中的異動交易(Transaction) 記錄備份至其他的磁碟或是離線備份檔(Offline Storage) 中。



Recoverer (RECO)

主要用於處理分散式異動交易(Distributed Transaction) 環境下的資料回復。通常用於分散式資料庫中的失敗處理。



LOCK (LCK_n)

主要用於處理多個 Oracle Instance 之間的鎖定 (Lock)。

資料庫結構

(Database Structure)

- ❖ 一個 ORACLE 資料庫可分為：
 1. 實體結構是以作業系統(Operating System)觀點去看 ORACLE 資料庫結構，主要是作業系統中 ORACLE 資料庫的所有檔案。
 2. 邏輯結構是以 ORACLE 資料庫系統的觀點去看 ORACLE 資料庫結構，主要是 ORACLE 資料庫系統運作的結構。



實體資料庫結構的構成

1. Redo Log Files(異動交易記錄檔)
2. Control File(控制檔)
3. Data File(資料檔)



Redo Log Files(異動交易記錄檔)

- 主要記載使用者所執行的異動交易 (Transaction) 記錄，以便日後進行資料回復之用。



Control File(控制檔)

- 每個資料庫至少有一個 Control File。它儲存一些與資料庫相關的關鍵訊息，Oracle 資料庫透過 Control File 保持資料庫的完整性(Integrity)，與決定資料回復需使用那一個 Redo Log File。



Data File(資料檔)

- Data File 用以儲存所有的資料庫資料，Oracle 資料庫是由一個或多個 Data File 所組成。Data File 被 Oracle 格式化為 Oracle 區塊(Block)，以便管理檔案內的儲存空間。



邏輯資料庫結構的組成

- ❖ 表空間(Tablespace)
- ❖ 段(Segment)、區段(Extent) 與資料區塊(Data Block)



表空間(Tablespace)

- 一個資料庫被劃分為一個或多個邏輯單位，該邏輯單位稱為表空間(Tablespace)。



OFA的七種表空間與其用途

表空間類型	用途說明
SYSTEM	儲存資料字典的相關資料表格與視觀表格。它是在建立資料庫時自動產生的，所以它不具有重建性。是不可以被離線(Offline)的。
TEMP	儲存系統處理排序或整理時所用的過渡性暫時資料。
RBS	儲存系統執行 ROLLBACK 時所需的舊資料。
INDEX	儲存索引資料。
USER	可以用來儲存使用者資料的表空間。
DATA	此類的表空間用以儲存獨立的表空間，用以簡化檔案管理。
TOOLS	儲存一些 ORACLE 工具程式執行時所需的資料表格。



段(Segment)、區段(Extent) 與 資料區塊(Data Block)

- ORACLE 透過段(Segment)、區段(Extent) 與資料區塊(Data Block) 等邏輯資料庫結構來控制磁碟實體空間的使用。



段(Segment)

1. 資料段(Data Segments)：對於每個沒有簇集(Cluster)的表格都有一個資料段(Data Segment)。
2. 簇集段(Cluster Segments)：對於每個簇集(Cluster) 都有一個簇集段(Cluster Segment)。其中每個簇集段包含一個或多個被設計者簇集(Cluster) 在一起的表格，其目的是爲了改善存取效能。



段(Segment)(cont.)

3. 回復段(Rollback Segments)：ORACLE 系統有一個或多個回復段(Rollback Segment)。回復段(Rollback Segment) 儲存當異動交易(Transaction) 回復(Rollback) 所需要的資訊。
4. 索引段(Index Segments)：儲存個別的索引(Index)。
5. 索引段(Index Segments)：儲存個別的索引(Index)。



段(Segment)(cont.)

6. 暫存段(Temporary Segments)：主要用來做暫存工作區，尤其是對查詢結果的排序。系統管理者(DBA)在建立使用者(Create User)時可以指定暫存段，否則系統會內定為表空間 SYSTEM。



表空間(Tablespace)

- ❖ 一個資料庫劃分為一個或多個邏輯單位，該邏輯單位稱為表空間(Tablespace)。每一個表空間是由一個或多個資料檔(Data File) 所組成，該資料檔(Data File) 儲存表空間中全部邏輯結構資料。



建立表空間—CREATE TABLESPACE

此指令的作用是在資料庫中建立一表空間，其語法如下：

```
CREATE TABLESPACE <表空間名> ↵  
DATAFILE <檔案名> SIZE <檔案大小> ↵  
[<AUTOEXTEND_選項>] ↵  
[DEFAULT STORAGE <儲存格式> ] ↵  
[{ONLINE | OFFLINE } ] ↵
```



更改表空間指令—ALTER TABLESPACE

此指令修改已存在的表空間，主要功能如下：

- ✓ 增加或重新命名 DATAFILE (資料檔)。
- ✓ 修改預定的儲存參數。
- ✓ 使表空間上線(Online) 或離線(Offline)。
- ✓ 開始或停止備份。
- ✓ 允許或不允許寫入表空間。



其語法如下：

```
ALTER TABLESPACE <表空間名>
  [ ADD DATAFILE <檔案名> SIZE <檔案大小> ]
  [ RENAME DATAFILE <舊檔案名> TO <新檔案名> ]
  [ DEFAULT STORAGE <儲存格式> ]
  [ { ONLINE | OFFLINE } ]
  [ { BEGIN | END } BACKUP ]
```



刪除表空間指令 – DROP TABLESPACE

此指令將已存在的表空間從資料庫中刪除。
其語法如下：

```
DROP TABLESPACE <表空間名>↓  
[INCLUDING CONTENTS [CASCADE CONSTRAINTS]]↓
```

主要的初始資料庫帳號

使用者帳號↵	密碼↵	主要工作↵
<code>internal</code> ↵	Oracle↵	處理資料庫 Administrator 的工作，它並不是一個真正的使用者帳號而是 sys 帳號的別名，一般的情況下，我們並不直接使用它登錄系統。↵
<code>scott</code> ↵	<code>tiger</code> ↵	它是 Oracle 系統的使用者學習帳號，擁有 connect 與 resource 的角色，並擁有 emp、dept、salgrade 與 bonus 四個表格。↵
<code>sys</code> ↵	<code>change_on_install</code> ↵	處理資料庫 Administrator 的工作。ORACLE 9i 以後的版本會在安裝時要求變更密碼。↵
<code>system</code> ↵	<code>manager</code> ↵	處理資料庫 Administrator 的工作。ORACLE 9i 以後的版本會在安裝時要求變更密碼。↵



資料庫的啓動

資料庫的狀況有：

- ❖ 「關閉」(Shutdown)
 - ❖ 「已啓動」(Started)(Nomount)
 - ❖ 「已掛載」(Mounted)
 - ❖ 「開啓」(Open)
-
- ❖ `STARTUP [OPEN | MOUNT |NOMOUNT]`



NOMOUNT 的模式

- 資料庫的啟動的第一階段是讓使用者能成功在記憶體中開啓 **INSTANCE**。
- 在 **NOMOUNT** 的模式下，可以執行指令來建立資料庫，並且產生 **Control file** 來控制資料庫的運作，以完成建立資料庫的動作。



MOUNT的模式

- 這是資料庫啓動的第二階段，此模式的目的是讓已建立成功的資料庫能順利的由”開啓 INSTANCE” 模式(NOMOUNT)進入“MOUNT” 的模式。
- 在這階段，資料庫已能順利讀取 Control file、Redo log files 與 Datafiles 的資料。



OPEN 模式

- 這是資料庫啓動的最後一階段，在這階段使用者可以依其權限正常地存取資料庫的資料。



關閉資料庫

❖ 資料庫管理者在關閉資料庫時視狀況可以有三種模式：

1. SHUTDOWN NORMAL：當資料庫管理者執行
SQL> SHUTDOWN NORMAL;

或

SQL> SHUTDOWN;

要求關閉資料庫時，會等待線上的使用者把其工作處理完成並離線後，才依序地執行”關閉資料庫”、”卸載資料庫”與”關閉 INSTANCE”等程序。



關閉資料庫(cont.)

2. SHUTDOWN IMMEDIATE：當資料庫管理者執行

```
SQL> SHUTDOWN IMMEDIATE
```

這個指令的優點是可以節省資料庫關閉的等待時間，所以當資料庫的連線人數很多，如果使用 SHUTDOWN NORMAL 的指令需要等待很多時間才能正常關閉，因此可以利用 SHUTDOWN IMMEDIATE 的指令來加速資料庫關閉過程。



關閉資料庫(cont.)

3. SHUTDOWN ABORT：當資料庫管理者執行
SQL> SHUTDOWN ABORT

這種是使用暴力方式關閉資料庫。使用這種方式關閉資料庫的時機，是當使用“SHUTDOWN NORMAL”或“SHUTDOWN IMMEDIATE”都無法關閉資料庫時才用這指令，不然儘量少用。