

資料庫管理講義

第一回

60767A-1



社團法 人 考友社 出版發行

資料庫管理講義 第一回



第一講 資料庫概念.....	1
命題大綱.....	1
重點整理.....	2
一、資料庫管理系統架構.....	2
二、資料模式概述與實體關聯模式.....	9
三、階層式資料模式與網路式資料模式.....	30
精選試題.....	32

第一講 資料庫概念



一、資料庫管理系統架構

- (一) ANSI/SPARC 架構
- (二)外層
- (三)概念層
- (四)內層
- (五)資料定義語言與資料操作語言
- (六)資料獨立與資料相依
- (七)資料管理師與資料庫管理師的區別
- (八)資料庫管理師的功能
- (九)主從式架構
- (十)分散式處理
- (十一)三階層架構
- (十二) DBMS 的組成
- (十三) DBMS 的分類

二、資料模式概述與實體關係模式

- (一)資料模式
- (二)資料模式的組成
- (三)資料模式的類型
- (四)實體關係模式
- (五)關聯模式的限制
- (六)實體關係圖轉成關聯模式資料庫綱目
- (七)將 ER 圖轉為敘述句
- (八)將敘述句轉為 ER 圖
- (九) ER model 的問題
- (十)關聯類型的轉換

三、階層式資料模式與網路式資料模式

- (一)階層式資料模式
- (二)網路式資料模式



重點整理

在日常生活當中，需要藉助電腦來存取或更新資料。例如上網讀新聞、聊天、訂購商品時，這些動作的背後事實上都隱藏著相關資料的擷取和修改。所有的網上新聞、聊天記錄和商品交易記錄都存在於所謂的資料庫（Databases）裡；或者更確切來說，是電子資料庫（Electronic databases）。即使不上網，但到學校註冊、到圖書館借書、到大賣場購物，相關的資料也會被教務處人員、圖書館館員和大賣場收銀員由操作系統記錄在資料庫裡，以便將來的查詢和處理。

事實上一般人並沒有直接接觸資料，而是透過操作資料庫應用系統（database application systems）來取得或更新資料的。例如上到某個新聞網站，如中時電子報（<http://www.chainatimes.com.tw>），你是在跟該網站系統互動。可以想像的是該網站系統的背後，有一個龐大的新聞資料庫。同樣當你到大賣場購買商品結帳時，幫你結帳的收銀員敲打著她的收銀機，實際上是在操作一個商品交易系統，該系統的背後，一樣有一個相當大的資料庫，儲存著商品和交易資料。

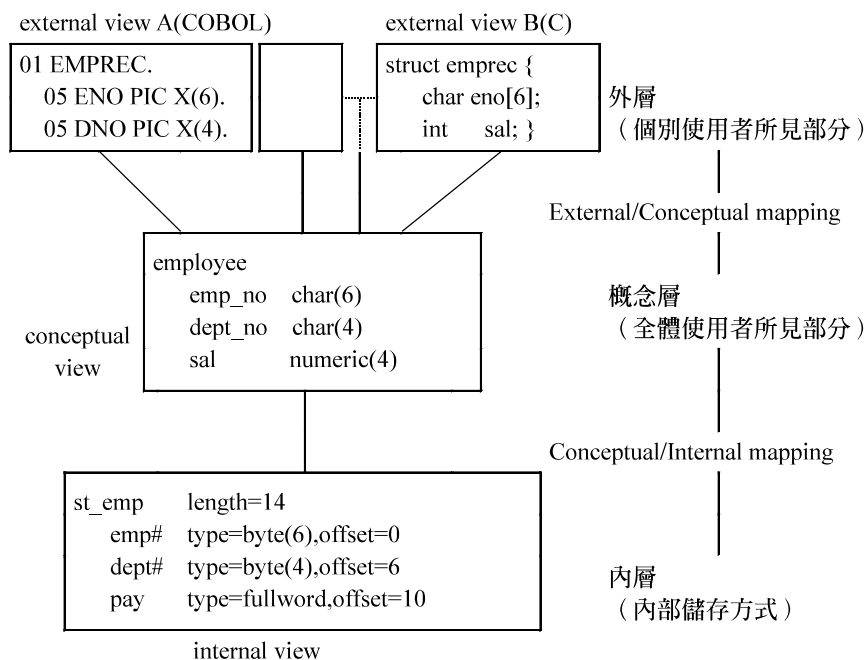
所謂資料庫就是一組相關資料的集合，乃是企業的應用系統所使用的一組不變的資料。此處所謂不變的資料是指並非一般短暫的資料，如輸入資料、輸出資料、控制指令等。

一、資料庫管理系統架構

資料庫管理系統是一組可以讓使用者建立與維護資料庫的程式，其架構如下：

(一)ANSI/SPARC 架構：

ANSI/SPARC 架構將 DBMS 分為三層，如圖(一)所示：



圖(-)

1. 內層 (Internal level) :

考慮資料的實際存放方式與存取方式，也就是資料庫的實體架構 (Physical structure)，考慮資料庫如何 (how) 存放。如圖(-)中即可表示 st_emp 在記憶體中各欄位的儲存長度與儲存方式。亦可表示是否建立索引來對資料檔案進行存取。每一個資料庫只有一個內層。

2. 概念層 (Conceptual level) :

考慮全體使用者看待資料的方式，也就是資料庫的邏輯架構 (Logical structure)，考慮資料庫內有什麼 (what)。如圖(-)中即可表示 employee 此一資料庫在綜合全體使用者的需求後，全體使用者所見部分。每個資料庫只有一個概念層。

3. 外層 (External level) :

考慮個別使用者看待資料的方式。如圖(-)中 external view A 即可表示某一 COBOL 程式所見部分，external view B 則可表示另一 C 程式所見部分。每個資料庫可能有一或多個外層。

(二)外層 :

1. 外層景觀 (External view) :

某一特定使用者所能見到的資料庫的內容。這是由許多類的外層記錄 (External record) 所組成，並由外層綱要 (External schema) 所定義。

2. 外層綱要 (External schema) :

利用使用者所用的 DDL，定義出特定使用者所用到的各種形式外層記錄，即可得到外層綱要，這是用來定義特定使用者的外層景觀。

(三) 概念層 :

1. 概念景觀 (Conceptual view) :

以較實際儲存形式抽象的方式所表示資料庫的整個內容資訊。這是由許多類的概念記錄 (Conceptual record) 所組成，並由概念綱要 (Conceptual schema) 所定義。

2. 概念綱要 (Conceptual schema) :

利用 DDL 定義出各種形式的概念記錄，即可得到外層綱要，若要保持資料獨立，則此時完全不可牽涉到任何儲存結構與存取技術。這是用來定義資料庫的概念景觀。

(四) 內層 :

1. 內層景觀 (Internal view) :

以較低階的實際儲存方式所表示的整個資料庫。這是由許多類的內層記錄 (Internal record) 所組成，並由內層綱要 (Internal schema) 所定義。

2. 內層綱要 (Internal schema) :

利用 internal DDL 定義出各種實體記錄，包括其儲存形式、索引是否存在、實際儲存順序等等，如此即可得到內層綱要。

(五) 資料定義語言 (Data Definition Language) 與資料操作語言 (Data Manipulation Language) :

1. DDL :

用來定義與宣告資料庫中的資料，通常依其定義對象分為 External DDL、Conceptual DDL、Internal DDL。

2. DML : 用來操作資料庫中的資料，通常包含在使用者所用的語言中。

(六) 資料獨立 (Data independency) 與資料相依 (Data dependency) :

1. 定義 :

資料獨立是指修改應用程式時，不必去更動資料架構，或者是修改資料架構但不必更動程式；反之即為資料相依。而依照程度又分為邏輯資料獨立 (Logical data independence) 與實體資料獨立 (Physical data independence)。

2. 邏輯資料獨立 (Logical data independence) :

是指修改概念綱要 (Conceptual schema) 而不需要修改外部綱要 (External schema) 與應用程式時稱之。

3. 實體資料獨立 (Physical data independence) :

是指修改內部綱要 (Internal schema) 而不需要修改外部綱要與應用程式時稱之。

(七) 資料管理師與資料庫管理師的區別 :

1. 資料管理師 (Data Administrator ; DA) :

是負責決定那些資料應存在資料庫中，並制訂與維護資料儲存政策，屬於高階管理人員。

2. 資料庫管理師 (Data base Administrator ; DBA) :

負責建立與維護實際的資料庫，並以技術控制方式強制推行 DA 的資料儲存政策，屬於資訊技術專業人員。

(八) 資料庫管理師的功能 :

1. 定義概念綱要 :

此一過程通常亦被稱為邏輯資料庫設計 (Logical database design)，當 DA 決定資料庫的內容後，DBA 就應利用 Conceptual DDL 產生概念綱要。

2. 定義內層綱要 :

此一過程通常亦被稱為實體資料庫設計 (Physical database design)，DBA 應利用 Internal DDL 產生內層綱要，同時也必須定義出內層綱要與概念綱要間的對應 (Mapping)。

3. 與使用者保持連絡 :

DBA 必須經常與使用者保持連絡，以確定其需求正確，同時也可以協助使用者寫出外層綱要，並且定義外層綱要與概念綱要間的對應。

4. 定義安全性與整合性規則 :

DBA 可以用 Conceptual DDL 定義出安全性與整合性規則。

5. 定義備份 (Backup) 與復原 (Recovery) 程序 :

為了確保資料庫的繼續正常運作，DBA 必須制訂備份程序，定期進行備份；並且要訂定當錯誤或故障發生後的復原程序，以減少組織資訊的流失。

6. 監督效能 (Performance) 並對修改的需要做回應 :

DBA 監督資料庫的效能，並在需要時對資料庫進行重組 (Reorganize) 以確保資料庫的效能保持在可接受的範圍內。另外也必須回應使用者修改的需求。

(九) 主從式架構 (Client/Server architecture) :

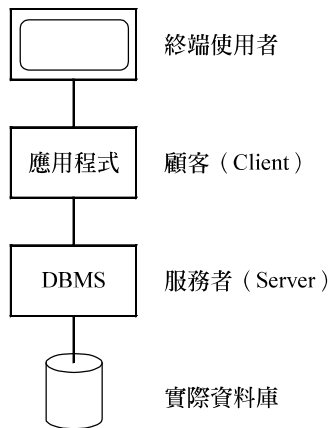
主從式架構分成兩個層次，如圖(二) :

1. 伺服器 (Server) 層 :

亦稱為後端（Backend），也就是 DBMS 本身，提供了前面所提到所有功能，包括內層、概念層與外層的功能，在主從式架構中扮演了接受顧客的請求，被動提供服務給顧客（使用者的應用程式）的角色，而不能主動提供服務。

2.顧客（Client）層：

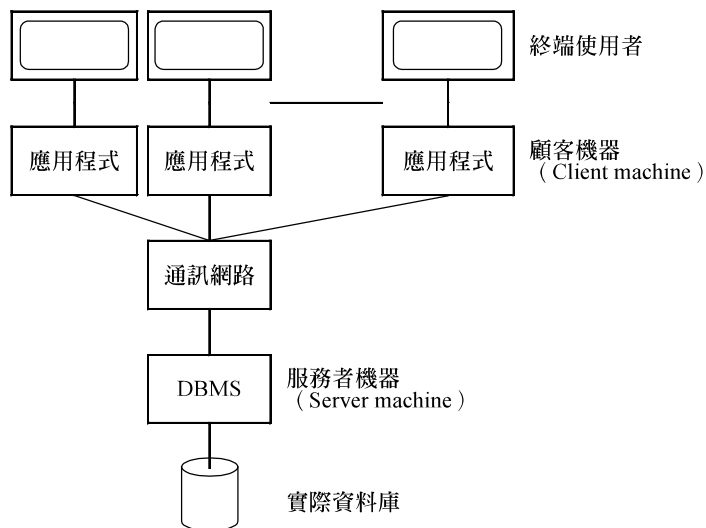
亦稱為前端（Frontend），也就是在 DBMS 上跑的應用程式，包含使用者自行設計的程式與外購的程式，均透過 DBMS 外層的介面與 DBMS 溝通。



圖(二)

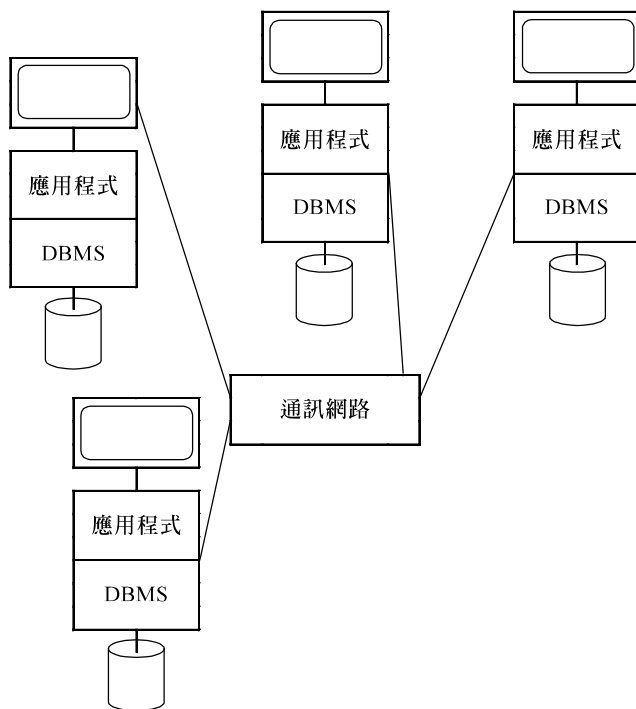
(+)分散式處理（Distributed processing）：

在主從式架構下，若前端與後端間是透過遠程通訊網路連接時，即成為分散式處理架構，這時可能伺服器只有一個，但有許多顧客，如圖(三)所示：



圖(三)

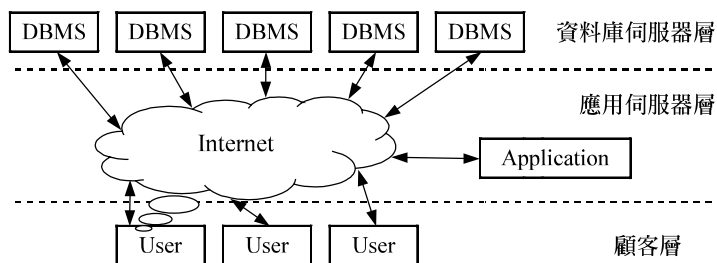
也可能顧客與伺服器在同一部機器上，同時提供顧客與伺服器的功能，可以讓其他機器上的顧客請求服務，如圖(四)所示：



圖(四)

(±)三階層架構 (3-tier architecture)：

採用三階層架構，如圖(五)所示，其中各種應用程式置於 Web server 上，擔任 Application server 的角色，提供各種企業邏輯 (Business logic) 功能；所需要的資料查詢則透過 Internet 連上網路上的其他 DBMS 進行，這些 DBMS 扮演資料庫伺服器 (Database server)，提供各種資料管理 (Data management) 功能；顧客端則採用瀏覽器 (Browser) 作為標準的使用者介面，提供各種展示 (Presentation) 功能。此架構是現在 Web DB 最常採用的架構，具有最大的彈性，讓系統管理者隨著技術的進步與需求的改變很容易可以單獨抽換各個階層。

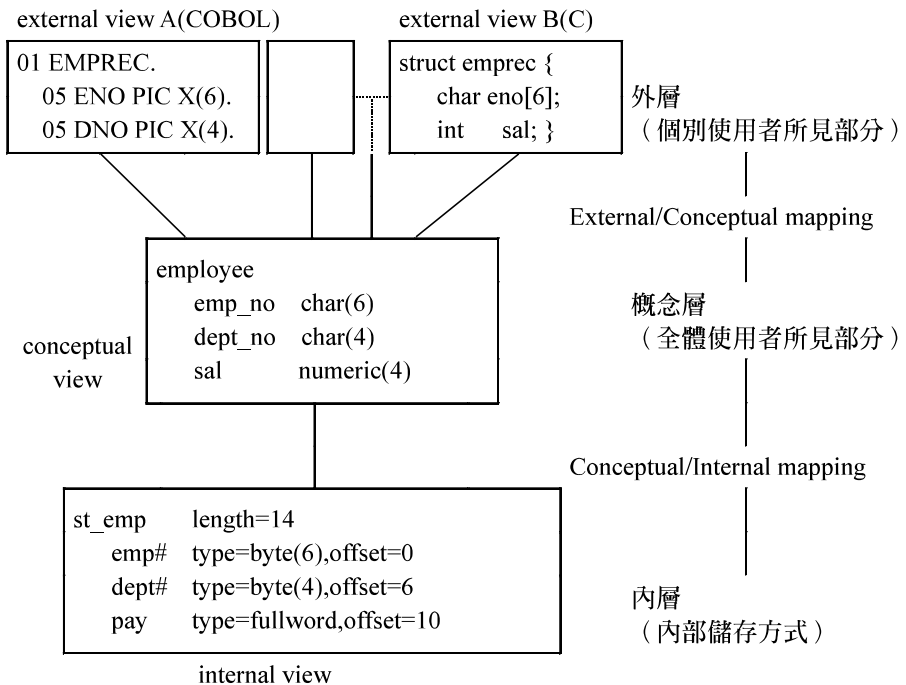


圖(五)

♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥
♥♥ 精選試題 ♥♥
♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥

一、請問 ANSI/SPARC 所定義之資料庫三層次架構與資料獨立性（Data independency）有何關係？請分別從邏輯資料獨立性（Logical data independence）及實體資料獨立性（Physical data independence）闡述。

答：(-)參見下圖所示：



1. 內層（Internal level）：

考慮資料的實際存放方式。如上圖中即可表示 st_eInp 在記憶體中各欄位的儲存長度與儲存方式。

2. 概念層（Conceptual level）：

考慮全體使用者看待資料的方式。如上圖中即可表示 employee 此一資料庫在綜合全體使用者的需求後，全體使用者所見部分。

3. 外層（External level）：

考慮個別使用者看待資料的方式。如上圖中 External view A 即可

表示某一 COBOL 程式所見部分（特別注意，個別使用者可能無法見到全部資料）；External view B 則可表示另一 C 程式所見部分。

(二)資料獨立與資料相依：

1. 定義：

資料獨立就是說修改應用程式時，不必去更動資料庫的架構；反之即為資料相依。

2. 邏輯資料獨立：

就是說修改資料庫的邏輯架構而不需要修改應用程式時稱之。

3. 實體資料獨立：

就是說修改資料庫的實體架構而不需要修改應用程式時稱之。

二、解釋下列名詞：

(一) Entity

(二) Relationship

(三) Weak entity type

(四) ER diagram

(五) Attribute

(六) Multivalued attribute

答：(一)實體（Entity）：

是指一個在真實世界中獨立存在的一個事物，可以是實際存在的物品，也可以是概念性的事物。例如公司、工作等。

(二)關聯（Relationship）：指描述實體類型間關係。

(三)弱實體類型（Weak entity type）：

實體類型中沒有鍵屬性時稱弱實體類型。例如員工親屬即為一弱實體類型，因某員工的親屬可能亦為另一員工親屬，因此無法找到鍵屬性。

(四) ER 圖（ER diagram）：用圖形的形式來表示實體關聯情形的圖。

(五)屬性（Attribute）：是指描述實體的一些特性。

(六)多值屬性（Multivalued attribute）：若一個屬性可能有許多值，則稱多值屬性。

三、試說明階層式資料模式與網路式資料模式間有何區別？並分析其利弊。

答：(一)階層式資料模式與網路式資料模式間區別：

1. 資料結構不同：

階層式資料模式記錄間僅能有階層式的親子關係，但網路式資料