



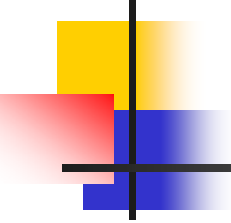
第一章 資料庫系統概論

黃仁鵬



1-1 資料庫系統概觀

資料庫(Database) 本身可視為一種電腦化檔案的儲存處所，並且是長期儲存於電腦中、有組織的、可分享共用的資料集合。



資料庫系統(Database System) ，提供一些工具給使用者存取或操作資料庫中的資料：

- 新增資料到資料庫中。
- 從資料庫中刪除資料。
- 從資料庫中更新資料。
- 從資料庫中查詢資料。



1-1-1 使用資料庫系統的優點

- 透過電腦化方式將資料集中控制與管理。
- 以集中方式來維護資料並且提供使用者存取或操作資料庫中的資料。
- 達到「資料獨立」。



1-1-2 資料庫系統的組成

- 使用者：使用者乃資料庫系統的主要服務對象。
- 資料：資料乃資料庫系統的主體。
- 軟體：在資料庫系統中實際儲存的資料與使用者之間的是資料庫軟體或資料庫伺服器(Database Server)。
- 硬體：資料庫系統的硬體包含電腦主機與存放資料的儲存媒體。

1-2 資料架構的發展



- 綱要(Schema)是資料庫中全體資料的邏輯結構和特徵的描述，它僅僅涉及到類型(Type)的描述，不涉及到具體的值。



無綱要架構

資料與程式並未分開，稱之為「無綱要架構」
(Non-schema Architecture)。

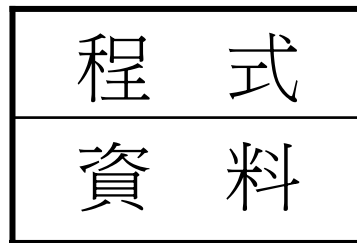


圖 1-2-1 無綱要架構

單一綱要架構

- 程式可以從檔案中讀入或是由鍵盤鍵入資料，只要確認外部資料的格式與程式所要的格式相同即可，也就是確立了「內部綱要」(Internal Schema)。

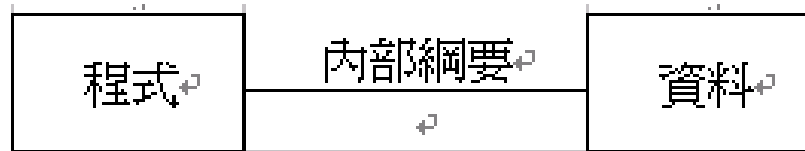


圖 1-2-2 單一綱要架構

兩層綱要架構

- 將使用者觀點與資料的儲存方式分開，因而產生了「外部綱要」(External Schema) 與「內部綱要」(Internal Schema) 的「兩層綱要架構」。

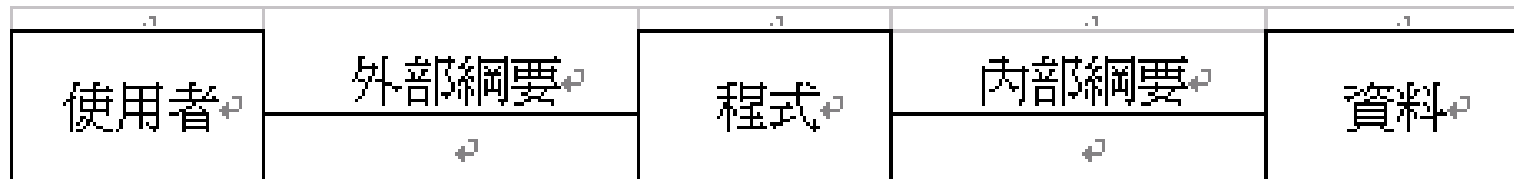


圖 1-2-3 兩層綱要架構



1-2-1 ANSI/SPARC 架構

它主要分成三層綱要：

- 外部層。
- 概念層。
- 內部層。



外部層(The External Level)

- 外部層代表使用者所看到的資料庫系統，會因為不同的使用介面與應用程式而不同。



概念層(The Conceptual Level)

- 它是外部層與內部層的溝通橋樑，代表資料庫設計者所看到的部份。



內部層(The Internal Level)

- 內部層主要是考慮資料的實際儲存結構與存取方式，例如：B⁺-tree，雜湊法(Hash)等。



映對(Mapping)

- ANSI/SPARC 架構有兩個映對關係
 - 概念層與內部層之間的映對 (Conceptual/Internal Mapping)
 - 外部層與概念層之間的映對 (External/Conceptual Mapping)。



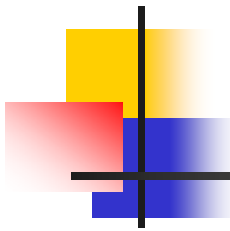
1-2-2 資料獨立 (Data Independence)

- 我們可以將「資料獨立」定義成「應用程式對資料儲存結構或存取方法的改變的免疫」。
- 也就是說，應用程式不會因為資料儲存結構或存取方法改變而必須更改，仍能照常運作。



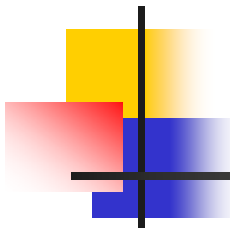
「資料獨立」種類

- 邏輯資料獨立(Logical Data Independence)
- 實體資料獨立(Physical Data Independence)



邏輯資料獨立(Logical Data Independence)

- 是指當概念綱要更改時，只需要DBMS更改外部層與概念層之間的映對以及概念層與內部層之間的映對，而不需要更改外部綱要、應用程式或概念綱要。



實體資料獨立(Physical Data Independence)

- 是指當內部綱要更改時，只需要DBMS更改內部層與概念層之間的映對，而不需要更改概念綱要或外部綱要。



1-2-3 資料庫管理者(DBA)

資料庫管理者的職責主要為：

- 定義資料庫的綱要。
- 監督並調整資料的效能。
- 安全性控制(Security Control) 與整合性檢查(Integrity Checking)。
- 資料備份(Backup) 與回復(Recovery)。



1-3 資料處理的演進

- 資料處理的演進主要是因為處理的工具與儲存媒體的進步和處理單元的不同，而分成不同的階段。



人工檔案處理

- 這是最早期的資料處理方式
- 利用卡片或紙張記載後歸檔
- 缺點：
 - 容易損毀
 - 處理統計或搜尋較費時間
 - 無法同時供多人查詢與使用
 - 所佔的檔案空間也日益增加
 - 多份複本造成資料重覆
 - 資料重覆造成資料的不一致
 - 資料不完整



電腦化檔案管理

- 使用電腦處理資料
- 此系統以檔案為處理單位
- 缺點：
 - 重覆性很高
 - 共用性很低



以記錄 (Record) 爲處理單元的 資料庫管理系統

- 檔案爲處理單元
- 缺點：
 - 在資料的處理應用和資料的一致性上仍有問題。



資料模式(Data Model)

- 階層式(Hierarchical)
- 網路式(Network)
- 關聯式(Relational)



階層式(Hierarchical) 資料模式

- 最早出現的資料模式
- 階層式資料庫系統是採用階層式資料模式做爲資料的組織方式。



階層資料模式的三個主要缺點

- 與現實世界不合
- 對於新增與刪除的操作限制較多
- 所有查詢必須通過父節點



網路式(Network) 資料模式

- 與階層式不同點在於一筆子記錄節點可以有零個至多個父記錄節點
- 允許一個以上的節點沒有父節點
- 允許兩個節點之間有多種關係(稱為複合關係)。



網路式資料模式優點

- 更能直接地描述現實世界，如一個人有多個親戚。
- 具有良好的性能與存取效能。



網路式資料模式的缺點

- 結構較複雜，而且隨著資料的增加，資料庫的結構就變得越來越複雜。
- 其 DDL (資料定義語言，Data Definition Language) 與 DML (資料操作語言，Data Manipulation Language) 複雜，使用者不易使用。



關聯式(Relational) 資料模式：

- 在關聯式資料模式中，資料與資料之間的關係不再是以指標為鏈結，取而代之是以資料值做為資料之間的關聯。



關聯式資料模式的優點

- 資料結構簡單易學
- 具有較高的資料獨立性。
- 它是建立於嚴謹的數學概念基礎上。
- 關聯資料模式的概念單一。
- 關聯式資料模式的存取路徑透明並具有更高的資料獨立性、更好的保密性，同時也簡化了資料庫開發的工作。



以物件為處理單元的物件導向式資料庫管理系統

- 物件導向式資料庫管理系統 (Object-Oriented DBMS)。
- 物件關聯式資料庫管理系統 (Object-Relational DBMS)。



1-4 資料庫管理系統的演進

- 集中處理式架構(Centralized Processing)
- 檔案伺服器架構(File-Server)
- 主從架構 (Client-Server)。

集中處理式架構

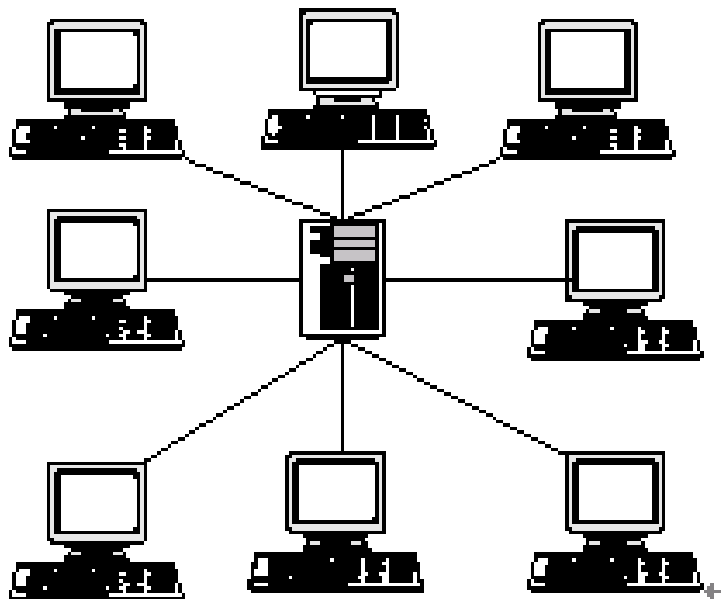


圖 1-4-1 集中處理式架構

檔案伺服器架構

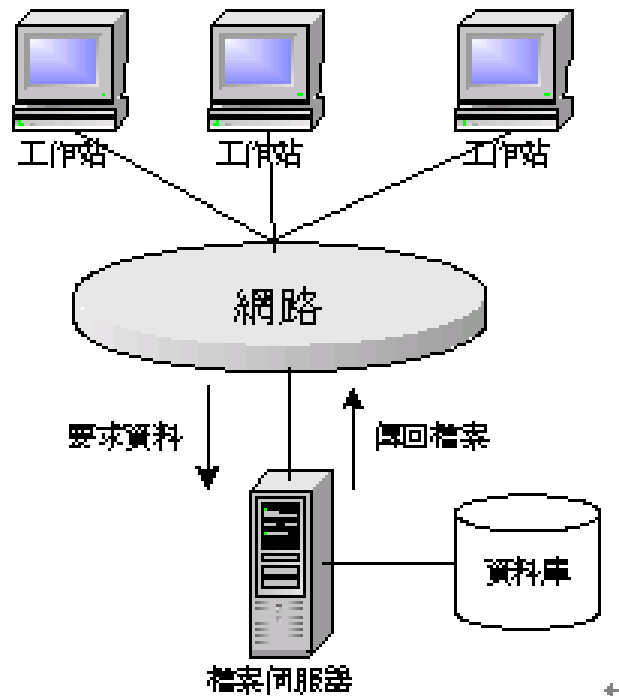


圖 1-4-2 檔案伺服器架構 ↵



檔案伺服器架構的缺點

- 會造成過多網路傳輸(Network Traffic)
- 每一個工作站都需要安裝資料庫管理系統
- 並行性(Concurrency)、回復性(Recovery)與整合控制(Integrity Control) 變得更複雜

主從架構

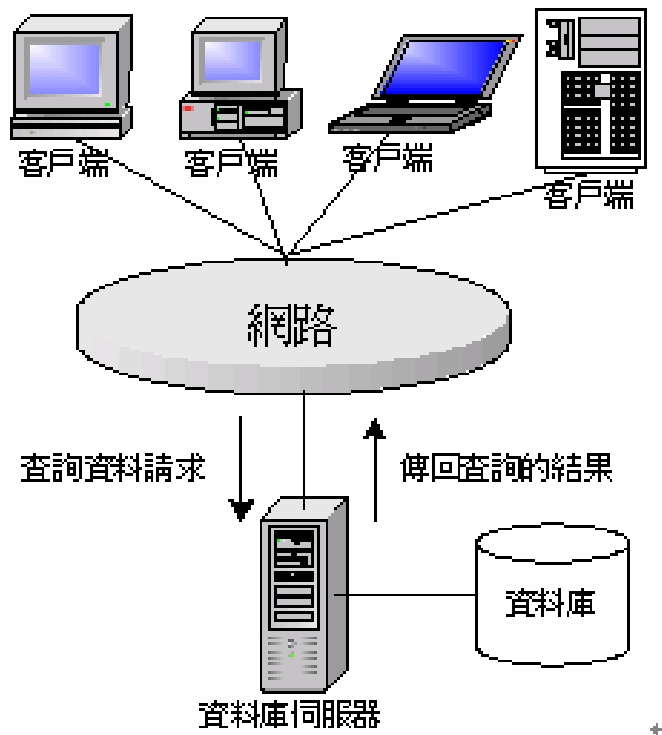


圖 1-4-3 主從架構 ↵



主從架構的優點

- 有效率地存取所有的資料庫資料
- 提高整體系統的效能
- 可以節省硬體系統的花費
- 可以減少網路傳輸的負荷。



主從架構的種類

- 單一客戶端，單一伺服器(Single Client, Single Server)。
- 多客戶端，單一伺服器(Multiple Clients, Single Server)。
- 多客戶端，多伺服器(Multiple Clients, Multiple Servers)。



1-5 資料庫規劃與設計步驟

1. 需求收集與分析(Requirements Collection and Analysis)。
2. 概念設計(Conceptual Design)。
3. 選用合適的資料庫管理系統(Choice of a Right DBMS)。
4. 邏輯設計(Logical Design)。
5. 實體設計(Physical Design)。
6. 系統實作(Implementation)。