

一個純粹網頁開發資料庫應用之研究— 以網際網路圖書管理系統為例

程裕繁 裘錦天
NantouBill@cttv.tinp.net.tw jchyou@mis.nccu.edu.tw
國立政治大學資訊管理學系
116 台北市文山區指南路二段 64 號

摘要

純粹使用網頁技術來開發資料庫應用具有技術整合與系統維護方面的優點。然而因為可供參考文獻與案例的缺乏、開發環境尚未充份支援視覺化開發等因素，目前 CGI、ISAPI 與 ASP 之類的網頁技術僅被部份用在資料庫應用的開發，對於比較強調使用者友善，而且沒有上網必要的部份，多數仍以 Delphi 或 Visual Basic 等工具來開發成視窗程式。本文則針對這些問題提供建議與解決方案。

系統資訊：網址：<http://books.NantouBill.idv.tw>，帳號：bill，密碼：@yesyes@

關鍵詞：全球資訊網、資料庫應用、使用者介面、視窗程式、網頁程式、班級圖書館

一、前言

對於會計系統、人力資源系統、進銷存系統……等資料庫應用的開發，在目前實務上的主流作法是，以 Delphi、Visual Basic 等工具來開發成視窗程式。

在另一方面，全球資訊網最初僅限於單向的多媒體資訊呈現。自從互動式網頁技術出現之後，全球資訊網已經逐漸被嘗試用來解決更為複雜的組織問題。目前的互動式網頁技術大致上可以區分為二種：一種為 CGI、ISAPI 與 ASP 之類，程式於伺服器端執行，並透過網頁伺服器的仲介，取得客戶端的輸入參數，然後將處理結果製成網頁送回客戶端呈現；另一種為 Java Applet 與 ActiveX Control 之類，嵌在網頁之中，並隨著網頁下載至客戶端電腦上執行。

目前雖然有使用網頁技術來開發資料庫應用的壓力（如提供網際網路上圖書檢索與預約機制的的需求等），可是網頁技術的使用仍然還不是資料庫應用開發的第一順位選擇，甚至於可以說是在不得已之下才會做的選擇。而為了能夠讓使用者透過瀏覽器來使用系統，實務上，多數系統同時採用了兩種技術來開發：有上網需求的部份用 CGI、ISAPI、ASP 等網頁技術來開發；沒有上網需求的部份則使用 Delphi、Visual Basic 等工具來開發。

同時使用兩種技術來開發系統，對於大規模的組織來說，他們需要分別聘用熟悉這兩種技術的人才，對於小規模的組織來說，他們的資訊人員可能就需要同時精通這兩種技術。如果將技術單純化，使用相同於建置網站的網頁技術來開發組織的資料庫應用，不祇不需要兩種專長的系統開發人員，更可以整合網站建置人員與系統開發人員，有效提升資訊人力資源的素質與取得成本。

在另一方面，相對於視窗資料庫應用，以網頁技術開發完成的網頁資料庫應用將具有下列三項優點 [2]：

（一）客戶端電腦除了作業系統與網頁瀏覽器之外，不需要再安裝其它軟體。並且，客戶端程式以網頁製作，集中存放在網頁伺服器內，由使用者依實際需要，動態下載至客戶端電腦上執行。解決了視窗資料庫應用棘手的客戶端安裝與維護問題。

（二）任何具備網頁瀏覽能力的電腦都可以執行客戶端程式。突破了視窗資料庫應用中，特定客戶端程式需要在特定電腦上執行的位置限制。

（三）客戶端程式以網頁瀏覽器為執行平台，與網路架構、電腦機型、作業系統的相關性，較視窗資料庫應用為低。具有較佳的跨平台能力。

目前已經有文獻探討如何純粹使用 Java Applet 或 ActiveX Control 之類技術來開發資料庫應用 [1,2,3]，但對於更廣泛被使用的 CGI、ISAPI 與 ASP 之類技術在資料庫應用上的探討就顯得不足。對於純粹使用 CGI、ISAPI 或 ASP 之類技術來開發資料庫應用，仍屬困難，也尚未被廣泛接受與採用，仍然需要解決兩個問題：

（一）可供參考文獻與案例的缺乏：目前幾乎沒有文獻在揭諸純粹使用此類網頁技術來開發資料庫應用的可行性及其優點，而教育如何純粹使用此類網頁技術來開發出完整資料庫應用的文獻也是一樣的缺乏。另一方面，可用來展現這種作法可行性與優越性並供開發者參考的實際開發案例也是付之闕如。

（二）開發環境尚未能充份支援視覺化開發：對於純粹以此類網頁技術製作資料庫應用來說，目前的 Dreamweaver、FrontPage 等工具祇提供了部份的支援。就如同 Delphi、Visual Basic 等工具出現之前的視窗資料庫應用開發一般，目前要開發出具有良好使用者介面的網頁資料庫應用仍然是一件困難的工作。

基於中區職訓中心電子商務班圖書館的成立與網頁程式設計教學上的需要，採用做做手段，純粹使用 ASP 這種最普及的網頁技術，完成了一個具有良好使用者介面的班級圖書館管理系統（<http://books.cvtc.gov.tw>，簡稱圖管系統）。由這個圖管系統可以顯示，妥善結合客戶端程式與伺服器端程式，確實可以做到相當接近視窗資料庫應用的設計。另一方面，在檢視圖管系統程式碼的過程當中，發現到每個相似功能之間的使用者介面與程式碼，其結構非常一致，而使用者介面與程式碼結構的一致性軟體開發自動化的前題之一。

本文主要呈現開發圖管系統所得到的心得與成果。先說明並示範如何做做視窗資料庫應用，純粹使用 ASP 網頁技術來開發出具有良好使用者介面的網頁資料庫應用，可供資料庫應用開發者參考。接著陳述圖管系統所呈現的一致性使用者介面與程式碼結構，可供視覺化工具開發者參考。

二、視窗資料庫應用的一般特徵

早期幾乎都是開發者使用特定程式語言，一行一行地撰寫出資料庫應用。這種開發方式不僅需要較高素質的開發人員，亦無法有效控制系統的品質與成本。隨著 Delphi、Visual Basic 等視覺化開發工具的進展，使用視覺化開發工具與現成的軟體元件，所視即所得地組裝出視窗資料庫應用的開發模式已經成為主流。而目前較為流行的視覺化視窗資料庫應用開發工具無論是在開發流程、基礎元件、或是在完成品的系統架構與使用者介面上都有相當一致的特徵，分別陳述如下：

(一) 在系統架構方面(參考以 UML [5] 類別圖所呈現的圖 1)：多數視窗資料庫應用採用二階式主從架構，由負責提供特定使用者所需功能的客戶端程式與負責資料管理的資料庫伺服器二種元素所組成。其中資料庫伺服器使用購買手段自供應商取得，而客戶端程式則自行或委外開發。在客戶端程式方面，使用者所負責維護的每項資料及其應取得的每項資訊，均應有其相對應的功能，好比某某資料的維護、某某資訊的查閱、某某報表的印列等。但在某些情況下，部份的功能或許會考慮予以省略，例如因為可以使用訂單維護功能來查詢訂單，所以可以省略訂單查詢功能。還有，雖然大部份的客戶端程式功能係衍生自資訊的產出需求與資料的維護需求，但仍有少許的功能係衍生自系統的限制，例如帳號管理功能即是衍生自系統的安全性要求。除此之外，某些功能之間具有高度的相關性，比如訂單維護功能與收據列印功能即是如此(因為收據祇是訂單資料的一種特殊觀點)，這些簡單的功能應該整合成一個具有高內聚力的複合功能。

(二) 在基礎元件方面：Delphi、Visual Basic 等工具都有一套元件庫(VCL 等)供開發者使用，以便能組裝完成所需要的視窗資料庫應用。在這些元件庫當中都有兩類與資料庫操作相關的元件。一類是資料集元件(Data Set)，為特定資料物件或 SQL 敘述執行結果的客戶端代理，能隱藏資料庫的多樣性與異質性，並提供一致的資料觀點、易用的操作方法、及簡單的資料異動驗證方法；另一類是資料感知(Data Aware)元件，為一種與特定資料集元件繫結(Binding)在一起的特殊使用者介面元件，會自動顯示所繫結之資料集元件的特定內容，並允許使用者對資料集元件進行瀏覽、修改、新增、刪除等操作。

圖 2 是這二類與資料庫相關元件的使用範例。圖中的視窗放置了資料集元件 A、資料瀏覽元件 B、資料方格元件 C、資料欄元件 D；A 是資料物件 E 的客戶端代理；藉由 A，使用者可對 E 進行存取；B 可供使用者改變 A 的目前記錄；C 可供使用者檢視 A 的多筆記錄，並可對 A 的目前記錄進行修改；D 可供使用者檢視與修改 A 的目前記錄的特定欄位。

(三) 在開發流程方面：前面已經提及，Delphi、Visual Basic 等工具提供開發者利用現成的軟體元件，以所視即所得的方式組裝出視窗資料庫應用，而這種軟體的開發方式稱為元件組裝開發模式。如以改良型 UML 活動圖 [4] 所呈現的圖 3 所示，這種開發模式將傳統的軟體開發活動切割成「元件工程」與「軟體工程」兩部份。其中，元件工程由元件提供者執行，專職特定應用領域的探究與分析，據以得出開發該領域軟體所需要的元件庫；而軟體工程則由軟體開發者執行，利用元件工程所發

展出來的元件庫，組裝出使用者需要的軟體。基本上，使用 Delphi、Visual Basic 等工具開發出來的視窗資料庫應用可以視為是一群相關視窗的集合，而其開發的過程是一種開視窗、放元件、設屬性、寫事件碼的過程，且這個過程是視覺化的、是再使用導向的、是複雜作業簡單化的。

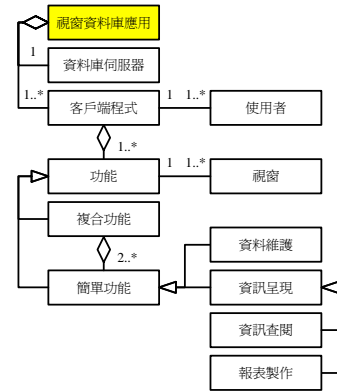


圖1 視窗資料庫應用的一般架構

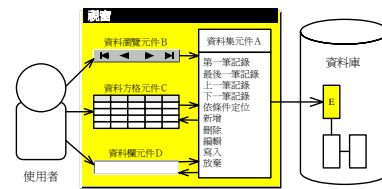


圖2 資料庫相關元件使用範例

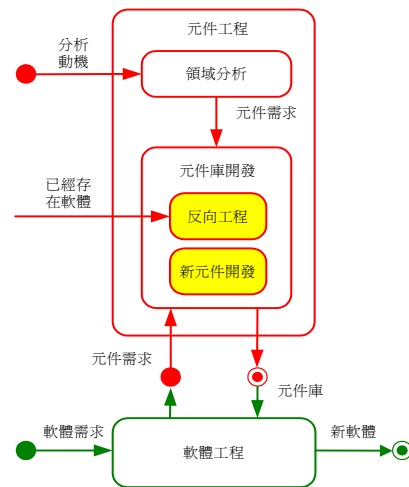


圖3 元件組裝開發模式

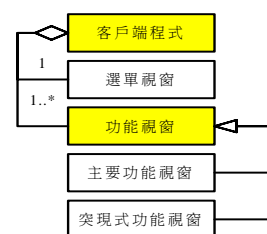


圖4 視窗角度的客戶端程式一般架構

(四) 在使用者介面方面：

1. 客戶端程式是一群相關視窗的集合，而每一種視窗負責一種責任。選單視窗陳列所有的系統功能供使用者點選，而功能視窗則提供使用者執行特定的作

業（參考以 UML 類別圖所呈現的圖 4）。客戶端程式由選單視窗開始執行，而當使用者點選陳列在選單視窗上面的功能之後，將會出現被選取功能的主要視窗，且在功能的執行過程中，隨著使用者作業需求上的變動與逐漸的精細，某些相關的突現式視窗也將依情況不定期出現。

2. 客戶端程式的功能是多視窗設計與先概約後進階設計的。也就是單一功能由多張相關視窗共同合作完成，而且先會陳列摘要性資料讓使用者能夠概約但快速地進行重點資料的瀏覽與檢索，然後再以開啟新視窗的方式讓使用者進行特定資料的更進階處理。
3. 提供良好的使用者輸入檢驗機制。而這些輸入檢驗機制可能很單純，也可能複雜到牽涉了資料庫的內容值。例如身份證字號以檢查碼來驗證，不涉及資料庫的內容。而判斷讀者編號是否已經被使用的檢查，就必需使用到資料庫的內容才能夠進行。
4. 提供了部份作業的自動化。例如提供希望輸入相同內容值表格欄位之間的自動複製。
5. 提供良好的線上說明機制，即時且有效地輔助使用者的操作。
6. 提供所視即所得的報表預覽列印機制，避免無意義的紙張浪費。
7. 支援多樣化的統計圖表展現，讓資料的展現更為美觀與有意義。

三、純粹 ASP 開發資料庫應用

尋找成功典範來做做，是落後者最有效率與最低成本的改造方法。而目前的視窗資料庫應用無論是在開發方式亦或是在完成品的使用者界面上都可以算得上是很成功的，是純粹網頁資料庫應用開發上的最佳做做對象。

而本節則探討如何使用 ASP 網頁技術來開發出，在使用者界面上相似於以 Delphi、Visual Basic 所開發之視窗資料庫應用的網頁資料庫應用：（一）說明一個純粹 ASP 開發資料庫應用的可行架構；（二）提出一些做做視窗資料庫應用的設計策略。

3.1 純粹 ASP 開發資料庫應用架構

ASP 網頁是一種內嵌兩部份程式碼的特殊網頁。其中有一部份會先行被執行的程式碼是 ASP 網頁獨有的，也是 ASP 網頁與傳統網頁最主要的差異所在。而這個部份以 JavaScript 結合 ASP 環境來撰寫，並由網頁伺服器來解譯並執行之，因為主要用來進行與資料庫有關的處理，所以本文稱之為資料庫程式碼；另一部份程式碼以 DHTML、CSS、JavaScript 三種語言撰寫，在資料庫程式碼被執行完畢之後，這部份的程式碼將隨即從網站伺服器下載至使用者電腦並由網頁瀏覽器解譯並執行之，因為多數與使用者介面相關，所以本文稱之為介面程式碼。

另一方面，雖然以 ASP 網頁技術開發完成的資料庫應用，其整個運作過程勢必需要網頁伺服器與網頁瀏覽器的介入，但是這兩者所扮演的都是一種如同作業系統般的環境角色，因此對於資料庫應用的開發者而言，可以假設其不存在，不需要予以考慮。

依據前一節所分析的視窗資料庫應用一般特徵與前幾段對於 ASP 網頁技術的簡單陳述，可以知

道，必需結合介面程式碼與資料庫程式碼才能夠做出相似於視窗資料庫應用的網頁資料庫應用。

圖 5 使用 UML 的類別圖來呈現，是一個整理出來的純粹 ASP 開發資料庫應用的可行架構，係修改自前一節提及的視窗資料應用架構（參考圖 4），而開發者可以據以開發出如同視窗資料庫應用般運作的網頁資料庫應用。

這個架構大體上與視窗資料庫應用架構相同，但仍有兩點不同：（一）客戶端程式改以網頁來實作，是一群相關網頁的集合。（二）客戶端程式的程式碼不再單純地在相同的電腦裡頭的相同行程裡頭執行。而是嵌在不同的網頁之中，並一分為二，依序且分別地由網頁伺服器與網頁瀏覽器來執行。

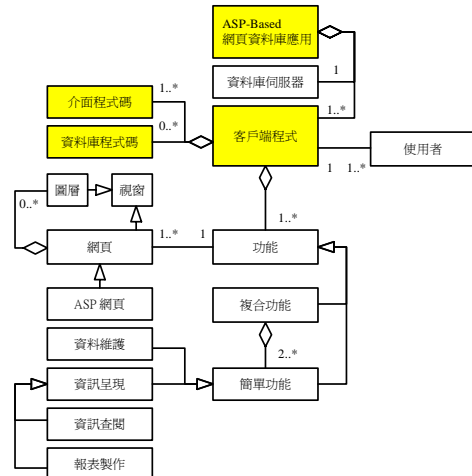


圖5 純粹 ASP 開發的資料庫應用架構

3.2 做做視窗資料庫應用的設計策略：

1. 採用前一小節所提出的架構，並將每張網頁的功能予以單純化。客戶端程式由選單網頁開始執行，並且每個功能由一張主要功能網頁與若干張的突現式功能網頁相互合作予以完成。
2. 在每張網頁的最前面加上登入要求程式碼，避免使用者繞過身份檢核機制，直接執行功能網頁。
3. 使用圖層或再開新瀏覽器實例（instance）來做做視窗資料庫應用所採用的單一功能多視窗的設計。而使用圖層來做製視窗會產生內容比較複雜的網頁，在撰寫與維護上比較困難；使用新瀏覽器實例來做製視窗則需要處理複雜的不同網頁之間的資料傳遞，並且也需要撰寫額外的程式碼才能夠保持回轉之後畫面與表單輸入內容的不變（參考圖 6）。
4. 使用超連結或 onClick 事件來進行視窗（圖層或新瀏覽器實例）之間的串接。而使用 onClick 事件可以做到更進階的處理。
5. 使用分頁陳列方式提供使用者進行大量資料記錄的瀏覽，並輔以直接指定頁次、設定篩選條件、直接資料檢索後定位等機制，提供更便利的使用者介面（參考圖 7）。
6. 使用表單的核取方塊（check-box input）結合部份的介面程式來完成資料記錄的動態挑選機制（單選或複選，參考圖 6）。
7. 呼叫視窗使用表格的隱藏欄位（hidden input）或是查詢字串（HTTP query string）將參數傳遞給被呼叫視窗。而被呼叫視窗則使用 Cookies 或 Session

變數將傳回值回傳給呼叫視窗。

8. 以程式產生程式的方式讓介面程式碼擁有動態操作資料庫資料的能力：資料庫程式碼先從資料庫裡頭取出需要的資料記錄，然後再動態產生部份的介面程式碼（即產生內含這些資料記錄的陣列宣告），如此客戶端即可動態使用資料庫資料（參考圖 8）。
9. 因為可以動態使用資料庫資料，輔以適當的介面程式碼，即可完成以資料庫資料為基礎的使用者輸入驗證（參考圖 8）。
10. 在執行被呼叫視窗前，先將表單資料存入 Cookies 或 Session 變數中，那麼當被呼叫視窗結束並回轉回呼叫視窗時，仍然能夠保存呼叫前呼叫視窗表格的內容值。
11. 以資料記錄的主鍵值來當影像的主檔名，並用特定的資料夾來予以儲存，再加上動態產生的 HTML 標籤，可完成影像資料的動態顯示。另再輔以檔案上傳機制（ftp server 等）也可以完成影像的儲存功能。

四、網際網路班級圖書館管理系統

中區職訓中心電子商務班為了有效應用每期固定的書籍費，並輔助班上學生的學習與老師的教學，特成立班級圖書館。另一方面，為了建立網頁程式設計課程所需要的教學教材，並實驗純粹使用 ASP 網頁技術來開發完整資料庫應用的可能性，特別採用了 ASP 網頁技術來開發這個班級圖書館的管理資訊系統。

4.1 開發流程

為了進行班上新設圖書館的管理，我們閱讀了一些圖書館管理相關的文獻資料，並廣泛收集了各種機關與學校圖書館相關的運作流程與管理規則，以做為班上圖書館作業流程設計與管理規則制定上的基礎。

做好了上述的基本工作之後，我們開始思考我們的需求與特殊點，以及資訊科技在圖書館管理上的各種可能性，然後我們規劃並設計出一套以網際網路圖管系統為運作核心的作業流程。而這個作業流程設計過程的產出成果包含了一套 e 化的班級圖書館作業流程、一個配合這套流程運作需要而制定的圖書館管理規則，以及這套流程運作核心資訊系統的開發規格。

在作業流程設計完成之後，緊接著我們依據圖管系統的開發規格，設計並開發完成了一個純粹以 ASP 網頁技術實作的圖管系統。

而在圖管系統完成之後，我們平行進行了圖管系統使用上的教育訓練、圖書資料的編目作業、基本電子資料的建置等三項工作。而待這些前置工作都已經就緒之後，班上圖書館正試開始運作。截至目前的運作情況都甚良好。

4.2 圖書館管理流程

圖書編目流程、圖書流通流程、每日例行處理流程等是這個班級圖書館的三個最主要作業流程，分別說明如下：

（一）圖書編目流程從圖書的購入到整備完成並上架為止，涉及了圖書編整與電腦檔案建立兩項

作業：中區職訓中心每年招訓兩期的電子商務班，且每期都編列有固定的書籍費用，這筆費用是圖書館的主要經費來源。而由於圖書館的規模較小且經費的來源固定，所以圖書編目的作業量並不算大，而且作業時間是固定且集中的。每當有新購圖書到館，管理員即使用文書處理軟體印製圖書條碼、圖書流通註記卡等文件來進行圖書的整編。而在圖書整編作業完成後再使用圖管系統的功能進行圖書電腦檔案的建立。



圖6 多視窗的讀者維護功能



圖7 提供多元資料篩選機制的圖書維護功能



圖8 能以資料庫內容進行驗證的假期維護功能

（二）圖書流通流程從圖書的檢索到歸還為止，涉及了圖書的檢索、預約、預約的取消、借閱、借閱的續借、歸還等六項作業：圖書使用網際網路圖管系統來管理，可以提供每天二十四小時的全年無休服務。舉凡圖書的檢索、預約、預約的取消、借閱的續借等作業，都可以透過網際網路，於任何時間在任何地方使用圖管系統來執行。而圖書的借閱與歸還作業，則需於正常上課時間親洽管理員辦理。

（三）每日例行處理流程涉及了借書通知、還書提醒、欠書催收等三項管理員每天都必需執行的例行性作業：管理員固定於每天早上九點左右，依序執行這三項工作，藉以通知／提醒／催促讀者到館辦理圖書的借閱或歸還。

4.3 系統功能需求

圖管系統是這個班級圖書館運作的核心，而這

個圖管系統的目的地在於全面性支援上述圖書館相關作業流程的運作。而基於作業流程運作上的需要，這個圖管系統提供了下列三類共十個功能：

(一) 使用者登入：基於安全上的需要，系統設計有身份稽核機制。而這個功能要求使用者輸入帳號與密碼，然後據以判斷其身份，並決定使用者可以執行的功能。

(二) 讀者方面的功能：

1. 圖書檢索：提供讀者一個多元但簡易的圖書檢索功能，並且在符合預約條件的情況下，提供讀者對特定圖書進行預約。
2. 個人預約管理：陳列讀者的所有圖書預約記錄，並提供讀者進行特定圖書預約的取消。
3. 個人借閱管理：陳列讀者的所有圖書借閱記錄，並且在符合續借條件的情況下，提供讀者進行特定圖書借閱的續借。
4. 個人資料維護：提供已註冊讀者自行修訂個人相關的基本資料。

(三) 管理員方面的功能：

1. 圖書流通作業：這是圖書的流通台機制，提供管理者進行圖書的借出與歸還作業。
2. 每日例行處理：提供管理者進行借書通知、還書提醒、欠書催收等三項每天都必需執行的例行性作業。
3. 基本資料維護：提供管理者進行圖書、讀者、出版商、圖書類別、管理規則、例假日（即行事曆）等六項基本資料的維護（新增、刪除、修改、查詢）功能。
4. 公告維護：提供管理者進行圖書館重要事項的公告。
5. 統計與分析：提供管理者與班導師一種圖形導向的統計與分析機制，有助於各種圖書館相關決策的制定。

4.4 系統設計

從圖管系統的功能需求來看，圖管系統有兩組功能需求明顯不相同的使用者——讀者與管理者，而且管理者的人數非常少，祇有一至兩位。因此替讀者與管理者分別開發一個專屬的客戶端程式，從安全的層面上來看是比較理想的設計。然而為了教學上的需要，不得已捨棄這種比較理想的設計方案，而將兩組原本應該予以獨立分開的功能擺放在一起，讓進階的管理功能也曝露在所有的網際網路使用者面前。除此之外，也是因為教學上的考量，圖管系統相關的文件與資訊一樣放置在功能選單上，方便學生的使用與學習。

另一方面，圖管系統遵循上一節所探討的純粹網頁開發資料庫應用方面的架構與策略來開發。以下則依序重點說明圖管系統的資料庫設計與客戶端程式各個功能的設計。

4.4.1 資料庫的設計

依據圖管系統的功能需求，以及客戶端程式執行上的需要，整個資料庫的設計如以 UML 類別圖所呈現的圖 9 所示，由十一張資料表格所組成。另一方面，考量了圖管系統可能的資料量與資料量成長速度、資料庫伺服器的特性與可能的負荷量、教學上的需要等因素，圖管系統採用 MS SQL Server 2000 資料庫伺服器。

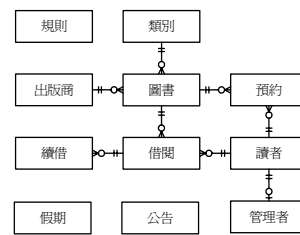


圖9 資料庫的實體關連圖

4.4.2 功能選單的設計

功能選單是圖管系統的起始點，由二張 ASP 網頁（內嵌有 ASP 程式碼的網頁）實作。其中一張主要用來陳列系統的所有功能與使用者該次連線的重要訊息，而另一張則內嵌在上述 ASP 網頁之中，用來陳列管理員所登錄的圖書館重要公告（參考圖 10）。

4.4.3 使用者登入功能的設計

這個功能使用功能選單網頁裡頭的圖層來完成使用者帳號與密碼的輸入表單，並以另外一支 ASP 網頁來檢核使用者的身份。而唯有通過身份稽核的使用者才能夠執行陳列在功能選單上頭的功能。更進一步地，祇有以管理者身份登入的使用者才能夠執行管理者專屬的系統功能（參考圖 11）。



圖10 功能選單的設計



圖11 使用者登入功能的設計

4.4.4 圖書檢索功能的設計

圖書檢索功能由四張 ASP 網頁實作。其中一張提供讀者設定圖書的檢索條件並進行檢索；一張則分頁陳列圖書檢索的結果，並提供讀者進一步執行圖書的預約與詳細內容檢視兩項作業；還有一張用來實際處理讀者的圖書預約；最後一張則提供讀者進一步執行特定圖書的詳細內容檢視（參考圖 12）。

4.4.5 個人預約管理功能的設計

這個功能由兩張 ASP 網頁實作。其中一張用來

陳列讀者的所有預約記錄，並供讀者進一步執行預約的取消；另一張則用來實際處理讀者的預約取消，並顯示處理結果予讀者（參考圖 13）。

4.4.6 個人借閱管理功能的設計

這個功能由兩張 ASP 網頁實作。一張用來陳列讀者的所有借閱記錄，並提供讀者進一步執行借閱的續借；另一張則用來實際處理讀者的借閱續借，並顯示處理結果予讀者（參考圖 14）。



圖12 圖書檢索功能的設計

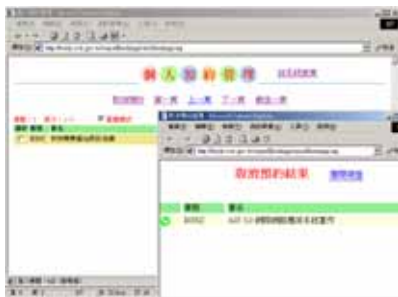


圖13 個人預約管理功能的設計



圖14 個人借閱管理功能的設計

4.4.7 個人資料維護功能的設計

這個功能由二張 ASP 網頁實作。一張提供讀者自行進行個人基本資料的修訂；另一張則實際更新資料庫裡頭的資料（參考圖 15）。

4.4.8 圖書流通作業功能的設計

圖書流通作業功能使用條碼閱讀器來自動化讀者與圖書資料的輸入，並以三張 ASP 網頁來實作。一張提供管理者進行借出或歸還資料的輸入；一張實際進行圖書的借出處理，並顯示處理結果予管理者；另一張實際進行圖書的歸還處理，並顯示處理結果予管理者（參考圖 16）。

4.4.9 每日例行處理功能的設計

通知借書、提醒還書、催收欠書等三個每日例行處理功能有著極為相似的設計。都有兩張 ASP 網頁，一張用來陳列應該進行 e-mail 寄發的所有資料；另一張則用來實際寄發 e-mail，並顯示寄發結果予管理者。比較特殊的是通知借書這個功能，多了一張用來製作明細表的 ASP 網頁，而這個報表則設計成使用網頁瀏覽器的列印功能來進行實際的印表機輸出（參考圖 17、圖 18、圖 19）。



圖15 個人資料維護功能的設計



圖16 圖書流通作業功能的設計



圖17 通知借書功能的設計

4.4.10 基本資料維護功能的設計

圖書、讀者、出版商、圖書類別、管理規則、例假日等六項基本資料的維護功能也有著極為相似的設計。都是採用兩層結構，先分頁陳列所有的基本資料供管理者進行瀏覽與刪除作業，然後再開新視窗供管理者進一步執行修改與新增作業。

另外，多數基本資料維護功能的修改與新增視窗使用圖層來完成，但圖書資料維護功能採用新增瀏覽器實例的方法來完成。這個不一致的介面設計情況將於圖管系統的下一個版本改善。

還有，讀者資料維護功能除了一樣有著資料的新增、刪除、修改、查詢等功能之外，另外又額外多出了借閱記錄檢視與預約記錄檢視兩項功能。（參考圖 6、圖 7、圖 8、圖 20、圖 21、圖 22）。

在實作上，圖書資料維護功能使用八張 ASP 網頁來完成，而其它的基本資料維護功能則祇使用了三張的 ASP 網頁來完成：



圖18 提醒還書功能的設計



圖19 催收欠書功能的設計

1. 資料篩選 ASP 網頁：這是祇有圖書資料維護功能才需要的 ASP 網頁。用來提供管理者設定圖書資料的篩選條件，藉以過濾掉不需要的資料，便於找到希望進一步維護的圖書資料。其它的基本資料維護功能則採用內嵌在資料陳列網頁的圖層來實作這個資料篩選機制。
2. 資料陳列 ASP 網頁：這是每一個資料維護功能都具備的網頁。用來依據管理者所建立的資料篩選條件，分頁陳列所有符合條件的資料供管理者執行進一步的維護作業。
3. 資料修改暨查詢 ASP 網頁：這是祇有圖書資料維護功能才需要的 ASP 網頁。用來提供管理者進行特定圖書資料的修改或檢視。其它的基本資料維護功能則採用內嵌在資料陳列網頁的圖層來實作這個機制。
4. 資料新增 ASP 網頁：這是祇有圖書資料維護功能才需要的 ASP 網頁。用來提供管理者進行圖書資料的新增。其它的基本資料維護功能則一樣是採用內嵌在資料陳列網頁的圖層來實作這個機制。
5. 儲存 ASP 網頁：這是每一個資料維護功能都具備的網頁。用來實際進行新增資料或已修改過資料的儲存。
6. 刪除 ASP 網頁：這是每一個資料維護功能都具備的網頁。用來實際進行特定資料的資料庫刪除動作。
7. 預約記錄檢視 ASP 網頁：這是祇有圖書資料維護功能才需要的 ASP 網頁。用來提供管理者進行特定讀者目前所有預約記錄的檢視。
8. 借閱記錄檢視 ASP 網頁：這也是祇有圖書資料維護功能才需要的 ASP 網頁。用來提供管理者進行特定讀者目前所有借閱記錄的檢視。

4.4.11 公告維護功能的設計

這個功能相當單純，用來提供管理者進行圖書館重要事項的公告。由兩張 ASP 網頁實作，其中一張陳列十個公告填寫欄位，提供管理者新增公告或變更原有公告的內容。而另外一張則用來實際儲存

管理者新增或變動過的公告資料。這些公告資料將由功能選單予以陳列在畫面左方的公告欄上面（參考圖 23）。



圖20 出版商維護功能的設計



圖21 圖書類別維護功能的設計



圖22 圖書館規則維護功能的設計



圖23 公告維護功能的設計

五、圖管系統的使用者介面與程式碼結構

對於網頁資料庫應用的開發來說，我們急需要一種類似 Delphi 或 Visual Basic 的視覺化組裝工具來簡化開發的困難度。而系統架構的制定，以及使用者介面與程式碼結構的標準化都是開發這種工具的重要前題。

第三節已經提出了一個純粹以 ASP 為基礎的資料庫應用可行架構，而本節將再說明圖管系統所呈現的一致性使用者介面設計與程式碼結構，供工具開發者參考。

5.1 圖管系統的使用者介面設計

分析圖 6 至圖 8、圖 10 至圖 23 等十七張圖管系統的執行畫面，可以看出整個系統的使用者介面有著一致的設計風格，而且相似功能之間的使用者介面，其結構更存在著高度的一致性。請參照這十七張執行畫面並參考 4.4 小節對於系統設計的說明來瞭解圖管系統的使用者介面設計。

5.2 圖管系統的程式碼結構

表 1 至表 7 陳列了部份圖管系統的程式碼結構。從這七張分析表可以看出來，整個系統的程式碼具有高度的一致性，而且相似功能之間的程式碼結構更是幾乎完全相同。

例如，圖書、讀者、出版商、圖書類別、管理規則、例假日等六項基本資料的維護功能都有的資料陳列、資料刪除、資料儲存等三張 ASP 網頁，其程式碼結構就是完全相同（參考表 2、表 3、表 4）；又例如，個人預約管理與個人借閱管理都有的資料陳列 ASP 網頁的程式碼結構也完全相同（參考表 5）；再例如，通知借書、提醒還書、催收欠書等三個每日例行處理功能都有的資料陳列與 e-mail 傳送 ASP 網頁，其程式碼結構也一樣是完全相同（參考表 6 與表 7）。

六、結論

本文探討純粹網頁資料庫應用的開發。首先進行視窗資料庫應用一般性特徵的分析；接著提出一個修改自視窗資料庫應用架構的純粹 ASP 開發資料庫應用可行架構，以及一組以 ASP 技術來做視窗資料庫應用的可行策略；然後說明一個採用本文所提架構與策略而開發完成的班級圖書館管理系統，藉以驗證這兩者的可行性；最後我們說明這個圖書館管理系統所展現的一致性使用者介面設計與程式碼結構，供工具開發者參考。

目前，我們正準備對本文所提及的系統架構、做做策略、使用者介面結構、程式碼結構等進行更深入與廣泛的探討與改良。

表1 功能選單的程式碼結構

1	例行處理		
2	處理連線訊息		
3	宣告 CSS		
4	建立函數群	1	使用者介面處理函數
		2	防止非法使用者執行函數
5	建立使用者登入視窗（圖層）		
6	顯示公告資訊		
7	陳列所有功能供使用者選擇		
8	定時更新連線時間		

表2 基本資料維護功能陳列網頁的程式碼結構

1	例行處理		
2	防直接執行		
3	宣告 CSS		
4	建立標題與資料瀏覽列		
5	前置處理	1	產生一頁資料量的資料陳列
		2	建立分頁的資料集物件
6	建立函數群	1	輸入稽核函數
		2	不當操作防止函數
		3	視窗做做函數

7	建立圖層視窗群		
8	陳列讀者資料	1	串接資料陣列
		2	陳列讀者資料
9	結束處理	1	完成資料陣列
		2	回存使用者的表單輸入內容
		3	關閉資料集物件

表3 基本資料維護功能刪除網頁的程式碼結構

1	例行處理
2	防直接執行
3	建立資料集物件
4	刪除指定資料記錄
5	關閉資料集物件

表4 基本資料維護功能儲存網頁的程式碼結構

1	例行處理
2	防直接執行
3	儲存表單內容供主網頁回存
4	建立資料集物件
5	更新／新增資料記錄
6	關閉資料集物件

表5 個人預約／借閱管理功能陳列網頁的程式碼結構

1	例行處理
2	防直接執行
3	建立內含特定讀者預約／借閱記錄的分頁資料集物件
4	建立使用者介面處理函數
5	陳列特定讀者的預約／借閱記錄
6	關閉資料集物件

表6 每日例行處理功能陳列網頁的程式碼結構

1	例行處理		
2	防直接執行		
3	建立標題與資料瀏覽列		
4	建立內含所有應通知讀者的分頁資料集物件		
5	建立函數群	1	使用者介面處理函數
		2	通知傳送函數（e-mail）
6	陳列所有應通知的讀者記錄		
7	關閉資料集物件		

表7 每日例行處理功能傳送網頁的程式碼結構

1	例行處理
2	防直接執行
3	建立內含所有應通知讀者的分頁資料集物件
4	依序寄發通知函
5	顯示寄發結果
6	關閉資料集物件

參考文獻

1. 施學琦、程裕繁，下載式全球資訊網企業資料庫應用之架構與開發流程，第四屆國際資訊管理研究暨實務研討會，1998。
2. 程裕繁，企業資訊系統開發—下載式架構與再使用導向方法之研究，雲林科技大學資訊管理系碩士論文，1999。
3. 程裕繁、施學琦，網頁式動態電腦管理教學系統之架構與開發流程，遠距教育季刊，第 12 期，1999。
4. 程裕繁、吳郁瑩，一個衍生自 UML 的企業流程模式化技術，第六屆國際資訊管理研究暨實務研討會，2000。
5. Fowler, M. and Scott, K., UML Distilled: Applying the Standard Object Modeling Language, Addison-Wesley, 1997.