

# 行政院國家科學委員會專題研究計畫成果報告

## 一個 Web 資料庫應用軟體之維護工具的研究與實作 The Study and Implementation of a Maintenance Tool for Web Database Applications

計畫編號：NSC 88-2213-E-009-010

執行期限：87 年 8 月 1 日至 88 年 7 月 31 日

主持人：王豐堅 國立交通大學資訊工程系所

### 一、中文摘要

Web 上的應用軟體與一般應用軟體最大的不同是：Web 資料庫軟體的程式模組幾乎是透過檔案來區分，一個軟體可能由數百個、甚至數千個檔案所組成，亦即整個軟體的作業流程及運作方法是分散在這些檔案上。因此，當程式設計師對部份檔案進行維護「異動」(change)時，軟體中其他受影響的檔案也需一併修正方可保持軟體的一致性。維護動作的主要問題之一就是無法有效的確認出哪些是受此異動所影響的檔案。本計劃製作一套可以支援維護動作的工具。我們定義一個可表達 Web 資料庫軟體規格的模型，此軟體模型能夠清楚的描述維護所需的軟體規格。而藉由「逆向工程」的技術可以由軟體所包含的這組檔案自動地擷取出模型中所需要的規格。然後，根據模型中所描述的規格，我們所提出的工具就可以在維護階段提供有效的功能與資訊來協助程式設計師進行維護的動作。我們也實際的使用一個 Web 資料庫軟體來評估此工具能否在維護階段時提供正確且完整的資訊。

關鍵詞：全球資訊網、資料庫、軟體工程

### Abstract

The primary difference between Web database applications and general applications is that the program modules of the former are separate based on files. Thus, a Web database application may consist of hundreds or even thousands files (web pages).

It causes that the operation procedures and logic of an application are distributed among these files. When programmer maintains (changes) some files of a Web application, those affected by the changes should be modified simultaneously to keep consistency. One major difficulty of the maintenance work is to identify these affected parts. In this project, we have designed and implemented a tool for assisting the maintenance of Web applications. A model is defined for representing the specifications of Web database applications. This model of maintenance can clearly describe the specifications, which are extracted automatically from the files via the reverse engineering technologies. Based on the specifications, a maintenance tool has been designed to provide useful functionalities and information to assist programmers. Besides, a practical Web database application was used to evaluate the tool.

**Keywords:** World-Wide Web, Database, Software Engineering

### 二、緣由與目的

網際網路(Internet)技術的演進使得連接廣域網路的電腦逐漸普及[1]，而全球資訊網(WWW: World-Wide Web)提供一個跨平台的資訊交流管道，加速了網際網路上的資訊與知識的交流。WWW 的應用軟體主要是透過網頁(Web page)讓使用者經由網際網路來操作。製作網頁通常是用

HTML 語言[3] 撰寫，以便定義在瀏覽器上可以用超文字結構(hyper-text structure)方式呈現的資訊[4] [5]，讓使用者很容易的瀏覽。資料庫是一般應用軟體中相當普遍且重要的一部份，但目前的 HTML 並未包含資料庫存取方式的描述。所以，為了在網頁上提供資料庫的功能，許多公司提供了特定的作法。例如，Microsoft 所提出的 Internet Database Connector(idc) 及 HTML Extension (htx)。這些擴充雖然讓網頁具有更強大的資料庫存取功能；但相對地，也使得網頁的製作更為複雜。

Web 資料庫軟體與一般軟體最大的不同是：程式模組幾乎是透過檔案來區分。因此 Web 上的軟體大都是由數百個、甚至數千個檔案所組成，導致整個軟體的作業流程及運作方法分散在這些檔案上[6] [7] [8] [9]。這樣的情形造成了下列幾個問題：

1. 程式設計師必須直接操作這種以檔案為基礎的 Web 資源，這與一般軟體的製作方法以「概念」(concepts)為基礎的方式不同。例如，結構化分析以功能及作業流程的概念來分析及設計系統。當概念對應成一組檔案時，這些概念就無法以一個實體來存取，且無法由檔案所組成的軟體來描述完整的設計概念[10]。
2. 另一方面，並未有工具針對 Web 資料庫提供軟體維護上的支援，現有的網頁製作工具（例如：FrontPage、NetObjects Fusion 等[11] [12]）以顯示網頁的超鏈結情況為主，當引進資料庫之後，單單使用超鏈結結構已經不足以應付種種維護上的動作。例如，某個資料庫表格(table)中某欄位(field)之型態，其長度原先為 20 字元，但是現在根據需要改為 30 個字元，從超文字結構中無法知道有哪些檔案需要修正。

3. 還有一個問題經常發生在製作網頁時：一般人常常使用複製的方式來撰寫新的網頁，不管是複製其中一部份或是進行修改的整個檔案，當重複修改這些內容時，非常容易造成網頁之間在整體操作或是內容上的不一致[10]。

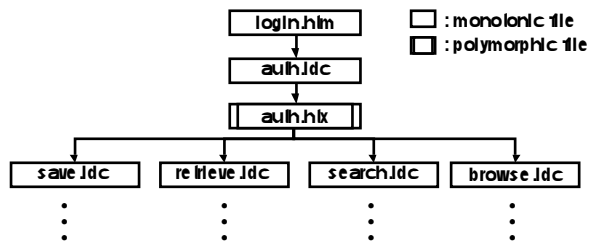
因此，當程式設計師對部份檔案進行維護「異動」(changes)時，軟體中其他受到影響的檔案也需一併修正方可保持軟體的一致性，而維護動作的主要問題之一就是無法有效的確認 (identify) 出哪些是受此異動所影響的檔案[13]。所以，就軟體的製作與維護而言，Web 資料庫軟體基本上比一般的軟體要困難的多。

### 三、結論與討論

本計畫的主要目的是針對 Web 資料庫軟體，製作一套可以支援維護動作的工具。而為了能讓程式設計師容易的進行維護的動作，我們所開發的工具可以讓程式設計師對於所維護的軟體有「概念」[14]，亦即程式設計師所面對的是軟體的規格，而由規格中可以清楚的明瞭所進行的維護動作會影響到原來的軟體的哪些檔案，或是哪些部份尚未進行修改。首先，針對 Web 資料庫軟體，我們提出一模型來描述系統的組織架構以協助軟體的維護。此模型主要包含三種圖示：超鏈結圖 (HLD: Hyper-Link Diagram)，實體關係圖 (ERD: Entity-Relationship Diagram)，與資料流程圖 (DFD: Data-Flow Diagram)。

超鏈結圖(HLD)是用來表示 Web 資料庫軟體透過網頁所呈現之超鏈結結構(hyper-link structure)，每個構成網頁的檔案由一個節點代表，節點與節點之間的有向線段表示超鏈結關係(可視為瀏覽的路徑)。節點又可區分為兩類：單調節點(monotonic node)與多形節點(polymorphic node)；單調節點是指其所代表的網頁在任

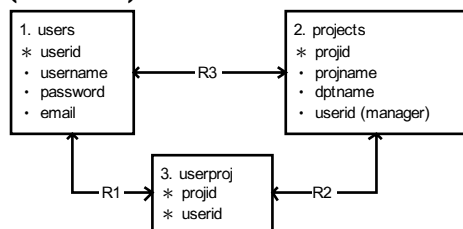
何情況下都顯示一樣的超鏈結結構，而多形節點是指這網頁所顯示的超鏈結結構視執行時所傳入的參數值決定。



圖一：超鏈結圖的範例

圖一是超鏈結圖的一個例子，每個長方形代表一個節點。在圖中表示 `auth.htx` 是一個多形節點，其餘皆是單調節點。此例子是一個應用軟體，它有四種功能（存入、取出、搜尋及瀏覽元件），而使用者有二種等級：會員可使用四種功能；非會員只能使用搜尋及瀏覽功能。所以圖中使用者藉由 `login.htm` 填入代號及密碼等資料後，透過 `auth.idc` 做身份的確認，確認結果再送給 `auth.htx` 後，`auth.htx` 會決定是要同時顯示四個功能的超鏈結或是只有顯示到 `search.idc` 及 `browse.idc` 這二個功能的超鏈結。在圖一中將節點分為單調與多形之後，可以讓程式設計師更清楚的明白使用者可能的導覽模式(navigation model)。

實體關係圖是用來表示關連式資料庫的規格，主要是由實體(entity)及關係(relationship)組成。關連式資料庫是由表格(table)組成，每個表格包含一組欄位(field)。在這裡的實體關係圖中，表格將會被視為實體，而欄位將被視為屬性(attribute)。

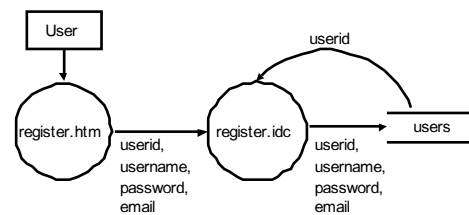


圖二：實體關係圖的範例

圖二是實體關係圖的範例，圖中共有三個實體 `users`, `projects`, `userproj` 及三個關係

`R1`, `R2`, `R3`，每個實體中所包含的屬性都直接列在實體名稱下面，屬性前面有星號“\*”標示代表此屬性是這個實體的主鍵(primary key 或 identifier)。例如實體 `users` 包括四個屬性 `userid`, `username`, `password` 及 `email`，其中 `userid` 是主鍵。

資料流程圖是用來表示應用軟體中資料的操作行為。一般的資料流程圖包含四種元素：外部實體(external entity)、資料儲存器(data store)、資料流(data flow) 及處理元(process)。在資料流程圖中的外部實體通常是使用者，資料儲存器可以定義為實體關係圖中的所用到的實體（即資料庫中的表格），資料流的內容是某實體內的屬性值（即表格中欄位的資料）在不同的網頁傳送關係，而處理元代表超鏈結圖中的節點(即程式檔案)。



圖三：描述使用者註冊動作之資料流程圖

資料流程圖的範例如圖三所示，圖三所顯示的是一個註冊動作的資料流程圖，首先處理元 `register.htm` 取得由外部實體 `User` 所輸入的四個資料流 `userid`, `username`, `password` 及 `email`。輸入的資料接著送到另一個處理元 `register.idc`，`register.idc` 會判斷輸入的 `userid` 是否已經存在於資料庫，所以有一條資料流 `userid` 由資料儲存器 `users` 流向 `register.idc`，若是使用者輸入的是尚未存在的 `userid` 時，所輸入的使用者資料將存入資料儲存器 `users` 中。由圖三的例子中，可顯示透過資料流程圖將可以容易地追蹤資料庫中各個表格及欄位在檔案中被處理的過程。

此軟體模型能夠清楚的描述維護所需的軟體規格。而藉由「逆向工程」的技術可以由軟體所包含的這組檔案自動地擷取出模型中所需要的規格，不需程式設計師

額外的輸入或設定。然後，根據模型中所描述的規格，我們所製作的工具就可以在維護階段提供有效的功能與資訊來協助程式設計師進行維護的動作[15] [16] [17] [18] [19] [20]。

#### 四、計劃成果

由於軟體的「維護階段」佔整個軟體發展至少 50% 以上成本，所以，如何增進維護動作的效率與正確性是非常重要的。而且，就目前在 Web 蓬勃發展的情況下，勢必會有更多的軟體開發是架構在 Web 之上，如何儘早設計出一個良好的 Web 資料庫軟體的維護工具是當務之急。本計劃製作一套可以支援維護動作的工具。我們定義一個可表達 Web 資料庫軟體規格的模型，此軟體模型能夠清楚的描述維護所需的軟體規格。而藉由「逆向工程」的技術可以由軟體所包含的這組檔案自動地擷取出模型中所需要的規格。然後，根據模型中所描述的規格，我們所提出的工具就可以在維護階段提供有效的功能與資訊來協助程式設計師進行維護的動作。就我們實際使用一個 Web 資料庫軟體來評估此工具能否在維護階段時提供正確且完整的資訊時發現：這套維護工具不僅讓 programmer 更能了解整個應用程式的邏輯，更增進了維護動作的效率與正確性。更進一部的資料，可參考相關已發表之論文[21] [22] [23]。

#### 五、參考文獻

- [1] J. Niesen, "Multimedia and Hypertext : The Internet and Beyond," AP Professional, San Diego, 1995.
- [2] "Internet and Intranet 1996: Markets, Opportunities, and Trends," Zona Research Inc., Redwood City, Calif., 1996.
- [3] I. S. Graham, "The HTML sourcebook: A Complete Guide to HTML 3.0," John Wiley & Sons, 1996.
- [4] M. Bieber et al., "Fourth Generation Hypermedia: Some Missing Links for the World Wide Web," International Journal on Human Computer Studies, to appear.
- [5] T. Isakowitz, "Hypermedia in Information Systems and Organizations: A Research Agenda," Proc. 26th Ann. Hawaii Int'l Conf. System Sciences, Vol. III, IEEE Computer Society Press, Los Alamitos, Calif., 1993, pp. 361-369.
- [6] D. Ingham, M. Little, S. Caughey, and S. Shrivastava, "W3Objects: Bringing Object-Oriented Technology to the Web," Fourth International World Wide Web Conference, Boston, Massachusetts, 1995, USA.
- [7] M. Bieber and T. Isakowitz, "Introduction to the Special Issue: Hypermedia in Information Systems and Organizations," J. Organizational Computing and Electronic Commerce, Fall 1996, pp. iii-vii.
- [8] F. Garzotto, L. Mainetti, and P. Palini, "Hypertext Design, Analysis, and Evaluation Issues," Comm. ACM, Aug. 1995, pp. 74-86.
- [9] P. Balasubramanian, T. Isakowitz, and E. A. Strohr, "Designing Hypermedia Applications," In Proceedings of the 27th Hawaii International Conference on Information Systems (HICSS), R. H. Sprau and B. Shriver, Eds. (Maui, Hi, Jan. 1994) IEEE Computer Society Press, pp. 354-365.
- [10] H. W. Gllersen, R. Wicke, and M. Gaedke, "WebComposition: An Object-Oriented Support System for the Web Engineering Lifecycle," The Sixth International World Wide Web Conference, 1997. URL: <http://proceedings.www6conf.org/HyperNews/get/PAPER232.html>
- [11] FrontPage Home Page, Microsoft Corp., URL: <http://www.microsoft.com/FrontPage/>
- [12] NetObjects Home Page, URL: <http://www.NetObjects.com/>
- [13] S.S. Yau and S.S. Liu, "Some Approaches to Logical Ripple-effect Analysis," Software Engineering Research Center, SERC-TR-24F, University of Florida, Oct. 1988.
- [14] S. Shlaer and S.J. Mellor, "Chapter 2: Information Modeling Concepts," Object Lifecycles: Modeling the World in States, Yourdon Press, 1992, pp. 11-31.
- [15] M. Bieber "On Integrating Hypermedia into Decision-Support and Other Systems," Decision Support Systems, 1995, pp. 251-267.
- [16] E. Rivlin, R. Botafogo, and B. Shneiderman, "Navigating in Hyperspace: Designing a Structure-Based Toolbox," Comm. ACM, Feb. 1994, pp. 87-108
- [17] R. Bptafogo, E. Rivlin, and B. Schneideman, "Structural Analysis of Hypertexts: Identifying Hierarchies and Useful Metrics," ACM Trans. Off. Info. Sys. 2, 10 (April 1992), pp. 142-180.
- [18] A. Diaz, and T. Isakowitz, "RMCCase:

- Computer-Aided Support for Hypermedia Design and Development,” In Proceedings of the International Workshop on Hypermedia Design. F. Grazotto, Ed. (Montpellier, France, June 1995)
- [19] R. Elmasri and S. Navate, “Fundamental of Database Systems,” 2nd ed. The Benjamin/Cummings, 1990.
- [20] F. Garzotto, P. Paolini, and D. Scgwabe, “HDM: A Model-Based Approach to Hypertext Application Design,” *ACM Trans. Off. Info. Sys.* 11, 1 (Jan, 1993), pp. 1-26.
- [21] H. Liao, J. Chen, C. Hsu, F. Wang, Li-Li Soung, ”支援軟體維護之 Web 資料庫軟體模型,” *National Computer Symposium*, Dec 22-23, 1997, vol. 3, pp. F56-F62
- [22] C. Hsu, H. Liao, J. Chen, and F. Wang, “A Software Maintenance Model for Web Database Applications,” *Fourth International Symposium on Autonomous Decentralized Systems*, Tokyo, Japan, March 21-23, 1999, pp. 338-344
- [23] Chen, J. -L., and Wang, F. -J., "Flow Analysis of Class Relationships for Object-Oriented Programs," accepted by *J. Information Science and Engineering*.