

國立交通大學

管理學院（資訊管理學程）碩士班

碩士論文



應用主題地圖於應用程式之資料庫綱要瀏覽
Applying Topic Maps to Navigate Application Database Schema

研究生：張家荃

指導教授：劉敦仁 博士

中華民國九十六年七月

應用主題地圖於應用程式之資料庫綱要瀏覽
Applying Topic Maps to Navigate Application Database Schema

研究生：張家荃

Student : Chia-Chuan Chang

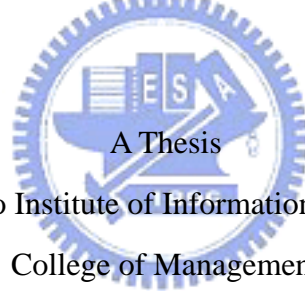
指導教授：劉敦仁 博士

Advisor : Dr. Duen-Ren Liu

國立交通大學

管理學院（資訊管理學程）碩士班

碩士論文



Submitted to Institute of Information Management

College of Management

National Chiao Tung University

In Partial Fulfillment of the Requirements

For the Degree of

Master of Science

in

Information Management

Jul. 2007

Hsinchu, Taiwan, the Republic of China

中華民國九十六年七月

應用主題地圖於應用程式之資料庫綱要瀏覽

研究生：張家荃

指導教授：劉敦仁 博士

國立交通大學管理學院(資訊管理學程)碩士班

中文摘要

由於資料庫管理系統龐大的儲存能力與精確且快速的處理速度，資料得以大規模地存入資料庫，使得資料庫的應用日漸普及，資料庫應用程式也因應而生。隨著資料庫愈來愈被廣泛使用，資料庫應用的軟體也愈來愈走向大型與複雜化，軟體間的競爭也提高許多，因此軟體開發的速度也得加快、不斷更新，而且品質要更好。

且自從 Internet 興起後，加上電子商務的出現，造成需求變更的速度加快，然而以往軟體開發的方式，是否能因應需求快速改變的環境，程式修改造成資料庫的變動，資料庫無法與應用程式完全契合，且當程式規模愈來愈龐大，開發人數眾多，長時間維護下新舊人員不斷更換，要怎樣維持軟體的開發與維護，達到最有效率的成果？

本研究嘗試從資料庫應用程式最基礎的來源資料—程式原始碼中擷取出影響資料庫資料異動的操作資訊，在大量的程式原始碼中，探勘出程式原始碼與這些資料操作之間關聯的資訊，並以視覺化的主題地圖的方式呈現其間的關係，有助於程式開發及測試人員更快速地了解資料庫應用程式與資料庫綱要之間的關係，達成高效率、高品質、低成本的軟體發展目標。

關鍵字：結構化查詢語言、XTM 語法、主題地圖、資料庫應用程式

Applying Topic Maps to Navigate Application Database Schema

Student : Chia-Chuan Chang

Advisor : Dr. Duen-Ren Liu

Institute of Information Management

National Chiao Tung University

Hsinchu, Taiwan, Republic of China

Abstract

With huger storage and faster processing speed in the database management system, the data will be stored massively. And it makes the database applications used popularly. Now the applications will be more and more complicated and large-scale, and competition among programers to develop the best system is intense. We need to develop the applications faster to keep our competitiveness.

The appearance of internet and e-commerce has brought about many changes in customers demand. We'll face more challenges to keep competitive . For example, will the database schema be consistent with the applications when causing changes to revise? How to keep the development and maintenance of the application systems to reach our goal when huger and huger scale, more and more programers involving, and faster and faster turnover rate in the software development ? Therefore, we need better method to shorten our developing time and promote the applications quality.

In the research, we try to retrieve the information of data manipulation which influences data variation from the most basic source of the database applications, that is, source codes. We try to extract the related information from large number of them, and transform the information into XTM to visulize the relationship using the knowledge navigation tool. We hope the developers and testers could understand the relationship between the application and database schema faster, and reach the software development goal of higher efficiency , higher quality and lower cost.

Keyword : SQL 、 XTM 、 topic maps 、 database application

誌 謝

感謝我的指導教授劉敦仁博士，劉老師認真研究與實事求是的精神，是學生良好的典範，在指導研究上，給予學生極大的自由空間，並適時地給予學生正確的指引，讓我能順利地完成系統與論文。

感謝交大資管所裡的老師和同學們，他們的指導與鼓勵，讓我能各方面有所成長。最後要感謝我的家人，因為他們的全力支持，我才得以安心的投注在課業上。

畢業並非結束，感謝這裡帶給我的一切，我會領懷學習到做研究的方法，邁向未來迎接新的挑戰，這是完成論文最大的收獲，我想這也是做論文的真諦。



目錄

中文摘要	I
英文摘要	II
誌謝	IV
目錄	V
圖目錄	VII
表圖目錄	VIII
1 緒論	1
1.1 研究背景及動機	1
1.2 研究目的	2
1.3 論文架構	2
2 文獻探討	4
2.1 結構化查詢語言	4
2.2 應用程式存取資料庫	7
2.3 主題地圖	9
2.4 XTM語法	11
3 系統架構概述	26
3.1 程式碼剖析引擎模組概述	26
3.2 XTM檔案匯出模組概述	26
3.3 網頁查詢過濾介面模組概述	26
4 系統模組說明	28
4.1 程式碼剖析引擎模組說明	28
4.2 XTM檔案匯出模組說明	34
4.3 網頁查詢過濾介面模組說明	37
5 系統實作與呈現	38
5.1 開發平台及工具	38
5.2 程式碼簡述及選取	38
5.3 資料庫綱要實作	47
5.4 程式原始碼剖析	48
5.5 XTM檔案匯出	53
5.6 視覺化主題地圖呈現	58
5.7 WEB查詢過濾介面	66

6	系統評估	68
6.1	原始碼與資料庫操作－「新增」之評估.....	68
6.2	原始碼與資料庫操作－「修改」之評估.....	69
6.3	原始碼與資料庫操作－「刪除」之評估.....	70
7	結論與未來研究方向	71
7.1	結論.....	71
7.2	未來研究方向.....	71
	參考文獻.....	73
	附錄 A：程式碼剖析及 XTM 匯出之原始碼.....	75
	附錄 B：Web 查詢過濾介面之原始碼.....	87
	附錄 C：XTM 檔案範例.....	91



圖目錄

圖 2.1 ODBC架構圖[20]	8
圖 2.2 XTM 發展歷程[2][17].....	10
圖 2.3 主題地圖概念實例[1].....	11
圖 2.4 XTM標籤結構圖[17]	13
圖 3.1 系統架構圖.....	27
圖 4.1 剖析原始碼中新增資料的資訊之流程圖.....	29
圖 4.2 剖析原始碼中修改資料的資訊之流程圖.....	30
圖 4.3 剖析原始碼中刪除資料的資訊之流程圖.....	32
圖 4.4 實體關係圖.....	34
圖 5.1 資料表關聯圖.....	47
圖 5.2 程式碼剖析使用者介面.....	48
圖 5.3 各程式語法設定資料表.....	49
圖 5.4 XTM匯出之使用者介面.....	54
圖 5.5 索引頁.....	59
圖 5.6 進入「程式碼」主題類別.....	60
圖 5.7 進入主題「MSSQL_MODIFY.ASPX」.....	61
圖 5.8 進入資源「MSSQL_MODIFY.ASPX(程式碼說明)」.....	61
圖 5.9 進入主題「PATENT」.....	62
圖 5.10 以「PATENT」為起點的視覺化主題地圖.....	63
圖 5.11 顯示與「PATENT」之關聯資訊.....	64
圖 5.11 透過EXPAND NODE顯示更多資訊.....	65
圖 5.13 顯示資料表「PATENT」的欄位.....	65
圖 5.14 程式碼與資料表關聯查詢.....	66
圖 5.15 詳細資料頁面.....	67
圖 6.1 PRECISION與RECALL示意圖.....	68

表目錄

表 1.1：原始碼中與執行時期擷取SQL敘述之比較	2
表 5.1：系統開發平台及開發語言與工具	38
表 5.2：實作的程式原始碼列表	39
表 6.1：原始碼與資料庫操作(新增)之評估	69
表 6.2：原始碼與資料庫操作(修改)之評估	69
表 6.3：原始碼與資料庫操作(刪除)之評估	70



1 緒論

1.1 研究背景及動機

從事軟體工程師迄今已有九年多，由於儲存成本逐年大幅度降低及儲存能力的提升，使得資料庫管理系統的應用愈來愈普遍，因此工作上也接獲到愈來愈多與資料庫應用有關的專案，例如經常接到客戶因應網際網路的發展，需將其系統改為網路版，以方便多人操作及管理，或因應客戶的需求擴充及變更，需要新增或修改資料庫的欄位。且軟體之間的競爭愈來愈激烈，如何在最短的開發時程，愈低的開發成本，達到更高的軟體品質，是軟體的發展目標，也是企業獲取更大利潤的基礎。

接到這樣的專案，開始思考是否有什麼方法可縮短開發的時程，讓開發的工作更有效率，如此便可降低成本，但又不失軟體品質，這才是利潤的泉源。資料庫應用程式，顧名思義是與資料庫有關的程式，因此若能更快、更完整地了解其間的關係，便能達成軟體發展的目標。

要了解應用程式與資料庫綱要之間的關係，可由資料庫應用程式最基礎的來源，即程式原始碼中，找出與異動資料庫資料相關的資料操作。在原始碼中與資料庫異動的資訊即潛藏於原始碼中的新增、修改及刪除的 SQL 敘述，欲擷取這些異動資料的 SQL 敘述，可於執行時期(runtime)擷取 SQL 敘述，然後直接剖析便可得到相關的資料表及欄位資訊。此方法得到的 SQL 敘述是百分之百的正確，但若要獲得所有的 SQL 敘述，需執行所有的可能的狀況，才能夠得到完整的資訊，因此需花費可觀的成本，且需於執行時期方可獲得此資訊，事先需將此系統安裝部署完成，但實務上，拿到欲修改的程式版本，常常無法安裝起來，不是沒有安裝說明，就是少了某些安裝元件。

另一個方法為，直接於原始碼中進行萃取異動資料的 SQL 敘述，此方法可有

效降低成本，並獲得完整的資訊，但其擷取的SQL敘述與執行時期得到的比較，其正確性較低，下表 1.1 為在原始碼中與執行時期擷取SQL敘述之比較：

項目\方法	原始碼中擷取 SQL 敘述	執行時期擷取 SQL 敘述
成本	較低	較高
效率	較佳	較差
正確率	較低	較高

表 1.1：原始碼中與執行時期擷取SQL敘述之比較

本研究採用原始碼中擷取 SQL 敘述的方式，嘗試由原始碼中擷取異動資料，即新增、修改及刪除 SQL 敘述，將相關的資訊，例如程式碼名稱、異動的資料表名稱、所在行數、此資料表的欄位等資訊存入資料庫，以利資料的交換。並將其間的關係，以視覺化的主題地圖方式呈現，俾使開發維護人員或新進人員，能夠更快速地了解程式與資料庫綱要之間的關係，進而縮短開發的時程，並提升軟體的可靠性及品質。



1.2 研究目的

使程式相關人員快速了解資料庫程式與資料庫綱要之間的關係，提高程式開發人員開發程式的效率與品質，及提昇測試人員測試之完整性，達成高效率、高品質、低成本的軟體發展目標，進而有效降低企業成本，提昇獲利能力。

1.3 論文架構

本論文共分為六個單元，第一單元為「緒論」，對研究背景及動機、目的以及論文整體架構做一簡單的說明。第二單元為「文獻探討」，介紹與本研究相關的主題。研究主題包含，結構化查詢語言、應用程式存取資料庫、主題地圖及 XTM 語法。第三單元為「系統架構概述」，簡介系統架構及主要模組。第四單元為「系統模組說明」，詳述架構中各模組的功能及其設計方法。第五單元為「系

統實作與呈現」，說明開發工具與平台、系統實作的方式與流程、與建置出的系統功能。第六單元為「系統評估」，說明評估系統的方法。第七單元為「結論與未來研究方向」，總結本研究的研究成果，並針對未來可能的研究方向加以說明。



2 文獻探討

2.1 結構化查詢語言

2.1.1 SQL 語言的興起與語法標準

SQL (Structure Query Language) 全名為結構化查詢語言，是與資料庫溝通的基礎，一切的指令對資料庫引擎而言，最終都要以 SQL 呈現[21]。SQL 語言是在 1970 年代晚期，由 IBM 公司在美國加州聖荷西(San Jose)的研究單位所發展出來的一套程式語言，當時是使用於 DB2 關聯式資料庫系統中。直到 1981 年，IBM 推出第一套商業用途的 SQL/DS 關聯式資料庫，再加上 Oracle 與其它業者也陸續推出許多種關聯式資料庫管理系統，才使得 SQL 語言被廣泛使用[19]。

SQL 的語法標準有「業界標準」與「ANSI SQL 標準」之分。關聯式資料庫百家爭鳴，各家公司都可能有各自的 SQL 語法或是定義不同的資料型別，為了避免各產品之間 SQL 語法不相容，形成多頭馬車的狀況，因此由 ANSI(American National Standards Institute, 美國國家標準局)制定出 ANSI SQL-92, 以定義出 SQL 的關鍵字與語法標準。其後更陸續發展出 SQL-99, 及目前最新版的 SQL-2003。大體而言，業界的產品都是包含 ANSI SQL 的基礎下，再擴充自家產品的功能，以求能展現出本身的特色[19]。本研究於程式原始碼中剖析的資料存取操作 SQL 語法，即新增(INSERT)、修改(UPDATE)及刪除(DELETE)等 SQL 語句語法[16]，本系統剖析的語法是屬於最基礎的 SQL 語法，各家對此都是使用相同的標準語法，詳細的 SQL 敘述將於以下 2.1.4(新增、修改、刪除的 SQL 語法)說明。

2.1.2 關鍵字、子句與敘述

SQL 語法的基礎是子句(clause)，子句中會包括一些關鍵字(keyword)。關鍵

字是對資料庫有特別意義的字，例如與存取資料相關的 SQL 關鍵字[15] 有”INSERT”、”INTO”、”UPDATE”、”SET”、”DELETE”、”FROM”、”WHERE” 等等。至於敘述(statement)則是指一組可產生存取資料庫結果的子句集合。例如：

UPDATE 員工資料表	這是一個 UPDATE 子句，使用到 UPDATE 關鍵字
SET 部門='研發部'	這是一個 SET 子句，使用到 SET 關鍵字
WHERE 員工姓名='張三'	這是一個 WHERE 子句，使用到 WHERE 關鍵字

以上三行子句組合起來便成為一個敘述，其作用是，更新(UPDATE)”張三”這名員工的員工資料表，將其部門改為”研發部”。

2.1.3 SQL 的分類

SQL 語法中的敘述，依用途可分為以下三大類[14]：

2.1.3.1 資料定義語言 (Data Definition Language, DDL)

凡是用來定義(或建立)資料庫物件，以及修改資料庫物件結構的 SQL 敘述，都屬於資料定義語言，包含了對資料庫及資料庫物件的建立與刪除指令以及更改表格及對表格加入整合性限制指令。例如包括 CREATE、ALTER、DROP 等命令的 SQL 敘述。

2.1.3.2 資料操作語言 (Data Manipulation Language, DML)

SQL 語法中用來做資料處理的敘述屬於資料操作這一類，即由一些對指定表格選取(SELECT)、新增(INSERT)、修改(UPDATE)及刪除(DELETE)四種命令所組成。而本研究所擷取的關鍵字即屬此類中會對資料造成異動的新增、修改及刪除三種命令。將於下一小節就這三種 SQL 敘述的語法做更詳細的說明。

2.1.3.3 資料控制語言 (Data Control Language, DCL)

一些專門用來設定資料庫物件(資料表、檢視表、預存程序等)使用權限的

SQL 敘述，屬於資料控制語言，即由一些控制資料安全、保密及並行處理的命令所組成。例如 GRANT(允許)、DENY(拒絕)、REVOKE(取消權限設定)等。

2.1.4 新增、修改、刪除的 SQL 語法

2.1.4.1 新增資料

INSERT 敘述的基本語法如下：

INSERT [INTO] tablename [(column list)] VALUES (data values)

tablename：欲新增的資料表名稱。

column_list：列出欲輸入值的欄位名稱，欄位名稱之間用逗號分隔。若沒有指定任何欄位，則表示資料表中的所有欄位。

data values：列出欲填入欄位中的值，值與值之間須用逗號隔開。此處必須與 column_list 互相對應。



2.1.4.2 修改資料

UPDATE 敘述的語法如下：

*UPDATE tablename SET {column_name={expression | DEFAULT | NULL}}[,...n]
[WHERE search_condition]*

column_name：欲更新的欄位名稱。

Expression：指定新的欄位值，expression 可以是一個常數、運算式、變數...等。

DEFAULT：可將 column_name 的欄位值重設為預設值。

NULL：可將 column_name 的欄位值重設為 NULL。

2.1.4.3 刪除資料

DELETE 敘述的語法如下：

DELETE [FROM] tablename [WHERE search_condition]

tablename：欲刪除的資料表名稱。

2.2 應用程式存取資料庫

資料庫管理系統提供以下兩種基本的技術供使用者在程式中使用：

2.2.1 應用程式界面

一種方式為使用者撰寫的程式，透過應用程式界面(Application program interface)的函數呼叫，與資料庫管理系統進行溝通。在程式中，使用者都是經由這一組 API 所提供的函數來對資料庫進行 SELECT、INSERT、UPDATE、DELETE 等操作。例如符合工業標準的 API—ODBC(Open DataBase Connectivity)界面。這樣的介面提供一個最大的操作空間，也就是一個應用系統經由一套共用的程式碼可存取不同的資料庫管理系統[18]。

ODBC 提供一組 API，使應用程式可以透過這一組 API 把 SQL 指令送到資料庫系統中存取資料。ODBC 是一個為資料庫系統定義的標準界面，因此應用程式可以同時與多個資料庫系統連接[20]。其架構包含應用程式(Application)、驅動管理員(Driver Manager)、驅動程式(Driver)及資料來源(Data Source)等四個部份，如下圖 2.1。

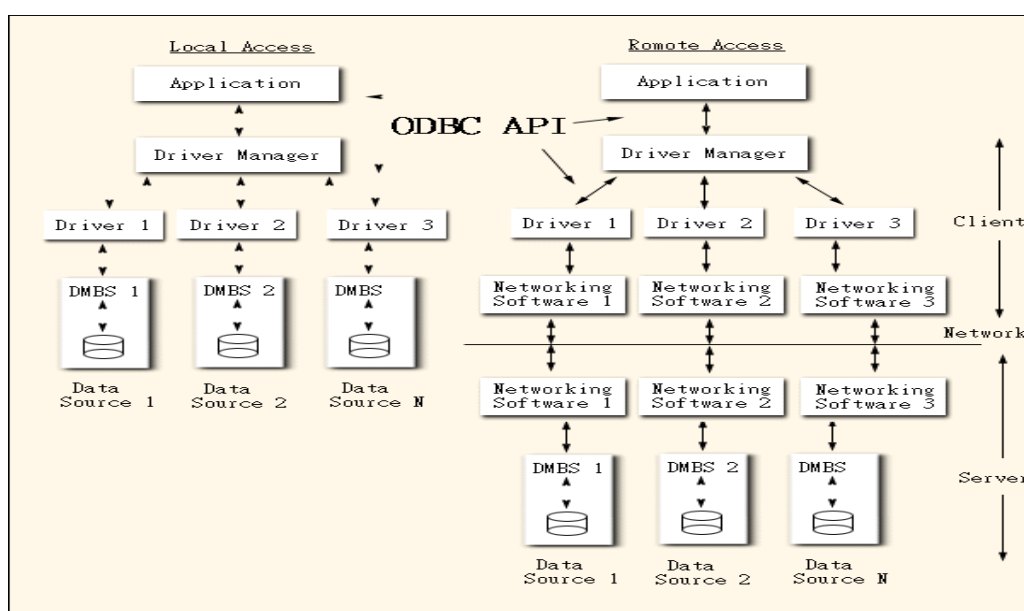


圖 2.1 ODBC 架構圖[20]

2.2.2 內嵌式 SQL

另一種方式為使用者直接在程式中輸入 SQL 敘述來存取資料庫，因 SQL 被當成是程式中的部分語法，所以我們稱這種方式為內嵌式 SQL(Embedded SQL, ESQL)。因大部份的程式語言無法直接辨識 SQL 的語法，因此在編譯之前，需要透過前置編譯器(Precompiler)先將 SQL 轉成程式可辨認的語法[13]。

以下為使用 C 語言的內嵌式 SQL 範例[8]：

```
//a C program segment that uses cursors with embedded SQL for update purposes.
prompt("Enter the Department Name: ", dname);
EXEC SQL
    select DNUMBER into :dnumber
    from DEPARTMENT where DNAME =:dname;
EXEC SQL DECLARE EMP CURSOR FOR
    Select SSN, FNAME, MINIT, LNAME, SALARY
    from EMPLOYEE where DNO =:dnumber
    FOR UPDATE OF SALARY;
EXEC SQL OPEN EMP;
EXEC SQL FETCH from EMP into :ssn, :fname, :minit, :lname, :salary;
while (SQLCODE == 0) {
    printf("Employee name is: ", fname,minit,lname);
    prompt("Enter the raise amount: ", raise);
    EXEC SQL
        update EMPLOYEE
        set SALARY = SALARY+ :raise
        where CURRENT OF EMP;
    EXEC SQL FETCH from EMP into :ssn, :fname, :minit, :lname, :salary;
}
EXEC SQL CLOSE EMP;
```

2.3 主題地圖

2.3.1 主題地圖的發展歷程

1980 年代晚期，幾位音樂家突發奇想，欲利用SGML發展出一種描述音樂時間特性的標準語言SMDL (Standard Music Description Language)，然而SMDL的發展卻因其可呈現時間觀點的特性而被美國中央情報局 (CIA) 看中，要求原本發展SMDL 的ISO 計畫停止在音樂上的研究，轉而將此計畫導向國防用途，結果促成了HyTime (Hypermedia/Time-Based Structuring Language) 的產生[11]。HyTime 主要應用於多媒體資訊的處理上，並提供定址及超連結的功能，於1992年成為ISO 17044 標準。

1991-1993 年間，Steven Newcomb主導了一個名為“Davenport Group”的研究小組，致力研究如何利用HyTime來呈現可互通的索引 (Indexes)、術語彙編 (Glossaries)、索引典 (Thesauri) 以及目次 (Tables of Contents) [2]。1995年，在圖像傳播學會 (Graphic Communications Association) 發起的HyTime應用大會 (Conventions for the Application of HyTime)上，Steve Newcomb 和Michel Biezunski 提出了一個新的構想，欲擷取書後索引隱含的知識架構，以促進索引的自動合併及改變其處理方式，並將這個構想定名為主題導覽地圖 (Topic Navigation Maps)，即為主題地圖的前身。

之後，主題地圖的概念不斷被充實與正規化，直到2000年，成為ISO 和IEC (International Electrotechnical Commission，即國際電子技術委員會) 聯合制定的國際標準ISO/IEC 13250:2000，此時的主題地圖是採用HyTime來定義其語法，稱為HyTM (HyTime Topic Maps)。然而在1998年，XML標準誕生了，這個由SGML 演化產生的簡易版本擷取了SGML的核心部分，卻更易於在網路上使用。隨著網路的普及，XML挾著強大的功能與彈性優勢，成為資訊流通的重要工具。因此主題地圖便由Newcomb 和Biezunski 創立的TopicMaps.Org 組織重新制定

採用XML 的新語法，稱為XTM (XML Topic Maps)，並於2002年被納入ISO 13250 標準中，XTM的發展歷程如圖2-2 所示。和HyTM相較，XTM 因使用較為方便，為大多數主題地圖的使用者所接受。

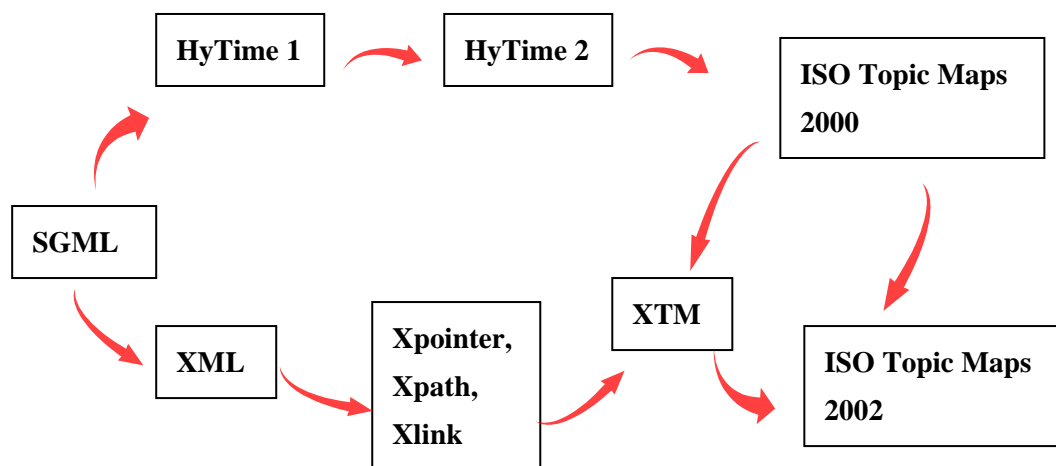


圖 2.2 XTM 發展歷程[2][17]

雖然XTM提供了建立主題地圖的語法，但在應用上仍需要其他相關標準的支援。負責維護ISO 13250 的委員會ISO SC34 已著手制定兩項與主題地圖相關的標準—ISO 18048 TMQL (Topic Maps Query Language) [4] 和ISO 19756TMCL (Topic Maps Constraint Language) [7]。TMQL是主題地圖的查詢語言，而TMCL則是定義主題地圖物件類別規則的語言。

2.3.2 主題地圖的內容概要

主題地圖包含了三個基本核心元素—T、A、O。T 代表Topics，也就是呈現某個概念的「主題」；A 代表Associations，表示各主題之間的「關聯」；O 代表Occurrences，翻譯成「資源指引」，提供和主題相關的資源連結[10]。當使用者對某個主題感興趣時，可以利用網路超連結的功能，直接連往相關資源進行閱讀，同時也能迅速找到和這個主題相關的其他主題，易於導引使用者在特定的資訊組織架構中探索。

圖2-3 是描述主題地圖基本概念的一個例子，圖中分成上下兩個區塊，上半

部簡單地表示醫院資訊管理架構，下半部代表和此領域相關的文件和資訊。在主題地圖中，各主題可依其性質，區分成不同的類別 (Type)。從圖2-3 的知識架構中，可發現三種主題類別：人、藥物和醫療器材，其中Topic 1 和2 代表「人」類別中的兩個主題，Topic 3 代表一種「藥物」，Topic 4 代表一種「醫療器材」。

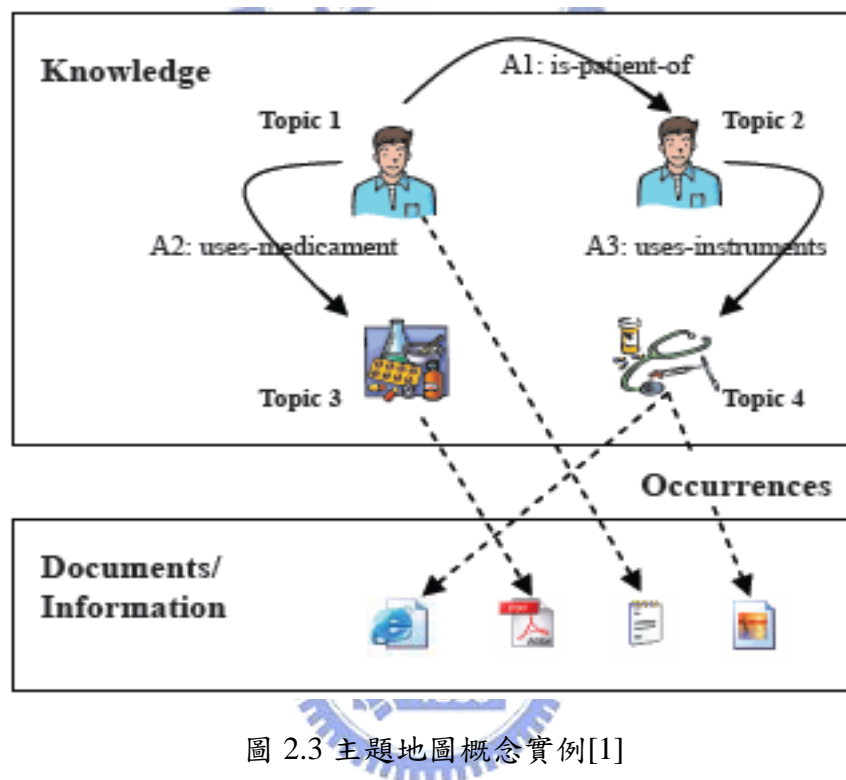


圖 2.3 主題地圖概念實例[1]

除了主題可歸納成不同的類別外，關聯和資源指引同樣包含許多不同的類別。如圖2-3 的四個主題之間包含三種關聯：A1 表示Topic 1 和Topic 2 之間的醫病關係，A2 表示Topic 1 服用Topic 3 這種藥物，A3 表示Topic 2 使用Topic 4 這種醫療器材。此外，在文件與資訊區塊中，我們可以看到各主題利用虛線指引和其相關的資源。

2.4 XTM 語法

XTM (XML Topic Maps) 是一種採用 XML 標記語言，由 TopicMaps.Org[12] 負責訂定，2001 年初發佈 XTM 第一版，2001 年 10 月，XTM DTD 被 ISO 13250 所採納[5]。XTM 目前已被廣泛用來定義主題地圖知識呈現架構的語法規範，總

共包含 19 個標籤 [6]。本章介紹 XTM 整體架構以及各部分的語法和用途，並以程式碼剖析引擎匯出的 XTM 檔案內容舉例說明。

2.4.1 XTM 整體架構

每一個 XTM 檔案中一定包含根元素 <topicMap>，用來指定這是一份主題地圖文件，<topicMap> 之下包含 <topic>、<association>、<mergeMap> 三個子元素，分別描述主題、關聯和主題地圖之間的合併。

在 <topic> 元素之下，包含 <instanceOf>、<subjectIdentity>、<baseName>、<occurrence> 四個子元素：<instanceOf> 指定主題所屬的主題類別；<subjectIdentity> 表示主題所代表的概念意涵；<baseName> 呈現主題的名稱；<occurrence> 指引和主題相關的資源。此類和 <topic> 元素相關的 XTM 語法將在第二節加以詳細說明。

在 <association> 元素之下，包含 <scope>、<member>、<instanceOf> 三個子元素：<scope> 指定關聯適用的範圍；<member> 標示關聯中包含的主題成員；<instanceOf> 指定關聯所屬的關聯類別。此類和 <association> 元素相關的 XTM 語法將在第三節加以詳細說明。

在使用 <mergeMap> 元素時，有些必須遵守的原則，我們將在第四節加以說明。圖 2.4 是 XTM 標籤結構圖，標示出所有標籤元素應該出現的位置，呈現 XTM 語法規範之架構全貌。

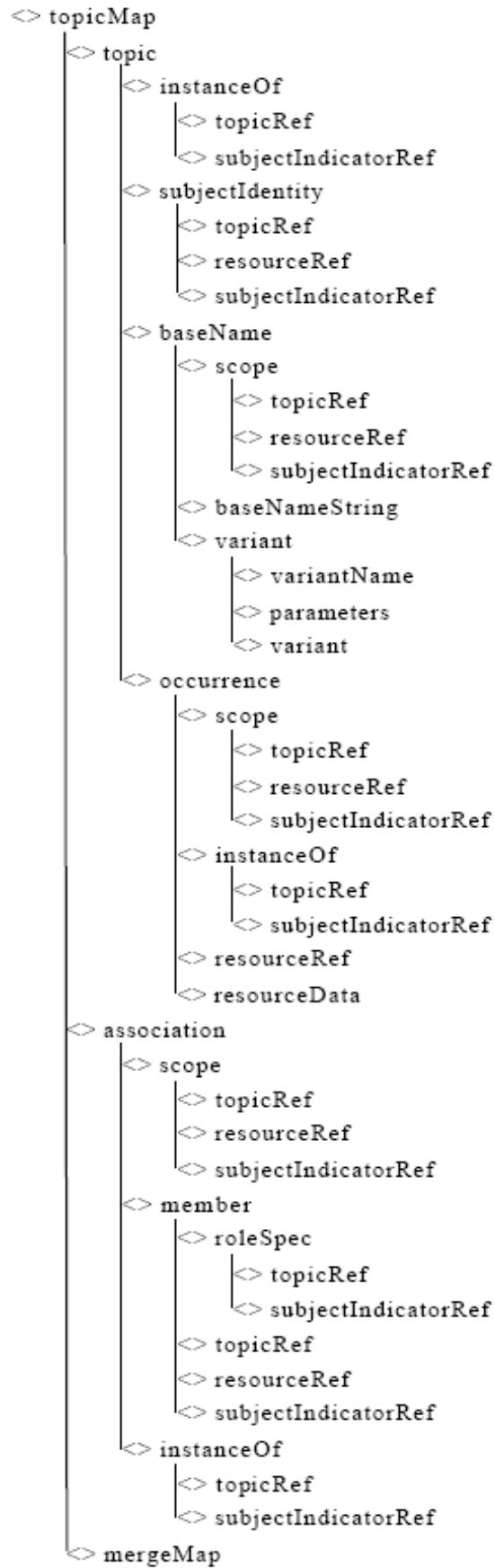
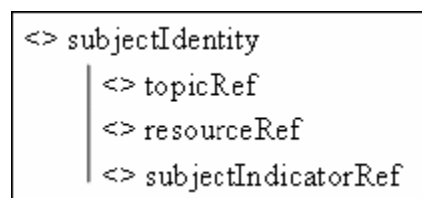


圖 2.4 XTM 標籤結構圖[17]

2.4.2 <topic>元素相關語法

2.4.2.1 <subjectIdentity>元素及其結構



<subjectIdentity>用來表達主題所代表的概念意涵，其下可利用<topicRef>、<subjectIndicatorRef>、<resourceRef>三個子元素來指引主題所代表的意義。

這三個子元素內皆具有Xlink 屬性 [3]，用來指引URI 連結。

<topicRef>連結的目標必須是主題，表示此主題和本身所描述的主題是同義詞；而<subjectIndicatorRef>則連結主題指引 (Subject Indicator)。主題指引是指能清楚代表主題概念的資源，為了方便主題地圖之間的交換與合併，明確的主題指引非常重要，一些重要的主題指引已被公開成為標準的公用資源，稱為“Published Subject Indicator”，簡稱PSI，例如“superclass”、“subclass”、“superclass-subclass relationship”表示階層關係的主題概念都已被制定成為PSI。<topicRef>和<subjectIndicatorRef>的差別在於<subjectIndicatorRef>連結的目標可為某個外部網頁，而<topicRef>則必須連結內部的主題或PSI。

另外，當資源本身即是主題所代表的概念時，我們用<resourceRef>來指引。

<subjectIdentity>的用法舉例說明如下：

```
<topic id="Source">
...
</topic>
<topic id="Source">
  <subjectIdentity>
<topicRef xlink:href="# Program"/>
  </subjectIdentity>
</topic>
```

上例中，主題“Source”藉由<topicRef> 指引其概念等同於另一主題“Program”。

```
<topic id="Table">  
  <subjectIdentity>  
    <subjectIndicatorRef  
      xlink:href="http://localhost/SCPE/#table "/>  
    </subjectIdentity>  
</topic>
```

上例中，主題“Table”藉由<subjectIndicatorRef>指引其概念為一PSI 所代表的意義。

```
<topic id="lettuce">  
  <subjectIdentity>  
    <resourceRef  
      xlink:href=" HTTP://LOCALHOST/Farm/images/lettuce.jpg"/>  
    </subjectIdentity>  
</topic>
```

上例中，主題“lettuce”透過<resourceRef>參考到一外部連結為程式碼的說明網頁，表示“lettuce”這個主題的概念就是圖片“lettuce(萵苣)”。



2.4.2.2 <instanceOf>元素及其結構

```
<> instanceOf  
  | <> topicRef  
  | <> subjectIndicatorRef
```

<instanceOf>用來指定主題所屬的類別。由於在主題地圖中，類別本身通常也被定義成主題¹⁶，所以<instanceOf>的作用即在表達主題與類別主題之間的階層關係。其下包含<topicRef>和<subjectIndicatorRef>兩個子元素可供選用，因前一小節已提及這兩個子元素的用法，此處不再贅述。舉例說明如下：

```
<topic id=" Source">  
  ...  
</topic>  
<topic id="SRC001">
```



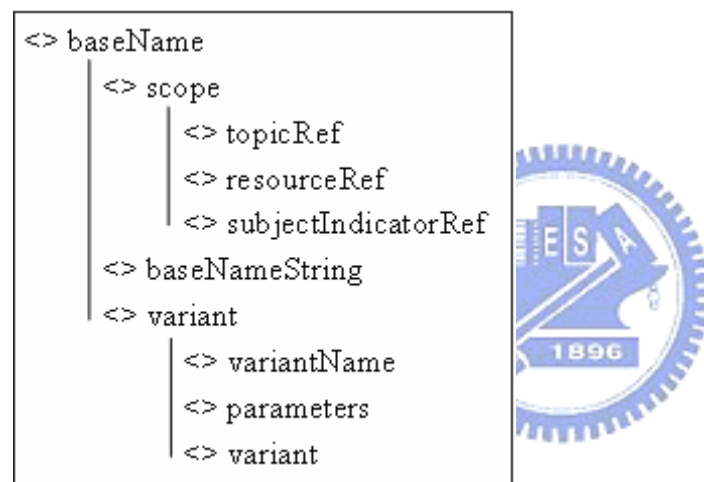
```

<instanceOf>
  <topicRef xlink:href="#Source"/>
</instanceOf>
<baseName>
  <baseNameString>程式原始碼名稱</baseNameString>
</baseName>
</topic>

```

上例中，主題“SRC001”（程式原始碼名稱）透過<instanceOf>參考到另一主題“Source”，表示“SRC001”屬於“Source”這個主題類別。

2.4.2.3 <baseName>元素及其結構



<baseName>用來呈現主題的名稱，其下包含<scope>、<baseNameString>和<variant>三個子元素。<scope>用來指示此名稱適用的範圍，透過<topicRef>、<subjectIndicatorRef>、<resourceRef>三個子元素參考到能表示其範圍的主題或概念；<baseNameString>導引出主題名稱字串；<variant>提供主題名稱的其他型式，利用其下的<parameters>指定<variantName>元素內名稱的用途，例如顯示用、排序用等。舉例說明如下：

```

<topic id="SRC002">
  <baseName>
    <scope><topicRef xlink="#current-name"></scope>
    <baseNameString>新的網頁抬頭名稱-修改日期</baseNameString>
    <variant>

```



```

<topic id=" NOTE001">
  <baseName>
    <baseNameString>原始碼001說明</baseNameString>
  </baseName>
  <occurrence>
    <instanceOf>
      <topicRef xlink:href="#RefData"/>
    </instanceOf>
    <resourceData>
      程式原始碼001說明文字
    </resourceData>
  </occurrence>
  <occurrence>
    <instanceOf>
      <topicRef xlink:href="#RefURL"/>
    </instanceOf>
    <resourceRef
      xlink:href="http://140.113.39.172/comic/OBJ0022_01.gif"/>
    </occurrence>
</topic>

```

上例中，主題“NOTE001”具有兩項相關資源，一項是文字介紹(RefData)，記錄於<resourceData>元素中；另一項則是外部網頁(RefURL)，透過<resourceRef>提供連結。

2.4.3 <association>元素相關語法

```
<> association
  |
  | <> scope
  |   |
  |   | <> topicRef
  |   | <> resourceRef
  |   | <> subjectIndicatorRef
  |   |
  |   | <> member
  |   |   |
  |   |   | <> roleSpec
  |   |   |   |
  |   |   |   | <> topicRef
  |   |   |   | <> subjectIndicatorRef
  |   |   |   |
  |   |   |   | <> topicRef
  |   |   |   | <> resourceRef
  |   |   |   | <> subjectIndicatorRef
  |   |   |
  |   |   | <> instanceOf
  |   |   |   |
  |   |   |   | <> topicRef
  |   |   |   | <> subjectIndicatorRef
```

<association>元素中包含三個部分：<scope>、<instanceOf>和<member>。

<scope>用來指定關聯適用的範圍，<instanceOf>用來說明關聯所屬的關聯類別，因前一節已提及這兩個元素的用法，本節僅針對<member>加以說明。

<member>用來描述關聯中包含的主題成員，此元素中可利用<roleSpec>來指定每個成員所扮演的角色，並以<topicRef>、<resourceRef>、<subjectIndicatorRef>加入主題成員。舉例說明如下：

```
<topic id="SRC001">
  <baseName>
    <baseNameString>原始碼一</baseNameString>
  </baseName>
</topic>
<topic id="TBL001">
  <baseName>
    <baseNameString>資料表一</baseNameString>
  </baseName>
</topic>
<association>
  <instanceOf>
    <topicRef xlink:href="#INSERT"/>
```

```

</instanceOf>
<member>
  <roleSpec>
    <topicRef xlink:href="#locate"/>
  <roleSpec>
    <topicRef xlink:href="#SRC001"/>
</member>
<member>
  <roleSpec>
    <topicRef xlink:href="#exit"/>
  <roleSpec>
    <topicRef xlink:href="#TBL001"/>
</member>
</association>

```

上例中的關聯利用<instanceOf>元素指定其關聯類別為「新增」(INSERT)，並透過<member>加入兩個成員，第一個成員的角色為「位於」(locate)，參考到的主題是原始碼一；第二個成員的角色為「存在」(exist)，參考到的主題是資料表一。



2.4.4 <mergeMap>元素相關語法

從名稱上便很容易看出<mergeMap>是用來將兩個不同的主題地圖予以合併，而合併必然會將數個相同的主題併成一個，連帶影響結合主題的關聯。在處理合併的問題時，有兩種合併的規則：[2]

1. 依據主題名稱的合併規則 (Topic Naming-constraint-based Merging Rule): 若兩個主題具有相同名稱 (baseName)，且其適用範圍 (scope) 亦相同，則這兩個主題會被合併。
2. 依據主題意涵的合併規則 (Subject-based Merging Rule): 若兩個主題具有相同的概念、意涵，亦即在<subjectIdentity>元素中指向相同的主題或資源時，則這兩個主題會被合併。

以下我們透過三個例子，分別說明合併的語法以及依據兩種合併規則進行合

併的結果。

2.4.4.1 合併的語法

以下是將兩個主題地圖加以合併的例子：

```
<mergeMap xlink:href="http://localhost/merge.xtm">
  <topicRef xlink:href="#person"/>
  <topicRef xlink:href="#place"/>
</mergeMap>
<topic id="person">
  .....
</topic>
<topic id="place">
  .....
</topic>
```

上例中，我們在<mergeMap>元素裡利用Xlink 指引出將要合併的外部主題地圖“merge.xtm”，並將位於目前主題地圖上的兩個主題——“person”和“place”加入“merge.xtm”的 scope，此舉的目的是為了釐清兩個主題地圖間彼此的 scope，亦即確認其“namespaces”，以避免因為「依據主題名稱的合併規則」而造成錯誤的合併。

2.4.4.2 依據主題名稱的合併

就合併而言，除了不同主題地圖間的主題合併，也有同一主題地圖中不同主題的合併。以下是同一主題地圖中依據主題名稱合併的例子：

[合併前]

```
<topicMap>
  <topic id="SRC001">
    <baseName>
      <baseNameString>程式碼一</baseNameString>
    </baseName>
    <occurrence>
      <resourceRef
        xlink:href="http://localhost/SRC/SRC001.aspx"/>
```

```

    </occurrence>
</topic>
<topic id="SRC002">
  <baseName>
    <baseNameString>程式碼一</baseNameString>
  </baseName>
  <baseName>
    <baseNameString>另一個名稱</baseNameString>
  </baseName>
  <occurrence>
    <resourceRef
      xlink:href="http://localhost/SRC/SRC002.aspx"/>
    </occurrence>
</topic>
<association>
  <member>
    <roleSpec><topicRef xlink:href="#Source"/></roleSpec>
    <topicRef xlink:href="#SCR001"/>
  </member>
  <member>
    <roleSpec><topicRef xlink:href="#NoteURL"/></roleSpec>
    <topicRef xlink:href="#NOTE001"/>
  </member>
</association>
<association>
  <member>
    <roleSpec><topicRef xlink:href="#Source"/></roleSpec>
    <topicRef xlink:href="#SCR002"/>
  </member>
  <member>
    <roleSpec><topicRef xlink:href="#NoteURL"/></roleSpec>
    <topicRef xlink:href="#NOTE001"/>
  </member>
</association>
</topicMap>
[合併後]
<topicMap>
  <topic id="SRC001">

```

```

<baseName>
  <baseNameString>程式碼一</baseNameString>
</baseName>
<baseName>
  <baseNameString>另一個名稱</baseNameString>
</baseName>
<occurrence>
  <resourceRef
    xlink:href="http://localhost/SRC/SRC001.aspx"/>
</occurrence>
<occurrence>
  <resourceRef
    xlink:href="http://localhost/SRC/SRC002.aspx"/>
</occurrence>
</topic>
<association>
  <member>
    <roleSpec><topicRef xlink:href="#place"/></roleSpec>
    <topicRef xlink:href="#SCR003"/>
  </member>
  <member>
    <roleSpec><topicRef xlink:href="#goods"/></roleSpec>
    <topicRef xlink:href="#NOTE001"/>
  </member>
</association>
</topicMap>

```

上例中，合併前的兩個主題“SCR001”和“SCR002”都包含「宜蘭縣」這個主題名稱，且因沒有指定scope，視為具有相同的適用範圍，故合併成為主題“SCR003”。而原本和“SCR001”及“SCR002”有關聯的主題“NOTE001”，則將關聯對象改成“SCR003”。

2.4.4.3 依據主題意涵的合併

以下是在同一主題地圖中，兩個具有相同意涵的主題進行合併的例子：

[合併前]


```

<topicMap>
  <topic id="SCR051">
    <subjectIdentity>
      <resourceRef
        xlink:href="http://localhost/SRC/SRC051.aspx"/>
      <resourceRef
        xlink:href="http://localhost/SRC/SRC052.aspx "/>
    </subjectIdentity>
    <baseName>
      <baseNameString>原始碼051</baseNameString>
    </baseName>
  </topic>
  <topic id="SRC052">
    <subjectIdentity>
      <resourceRef
        xlink:href="http://localhost/SRC/SRC051.aspx "/>
    </subjectIdentity>
    <baseName>
      <baseNameString>原始碼052</baseNameString>
    </baseName>
  </topic>
</topicMap>

```



[合併後]

```

<topicMap>
  <topic id="PER18">
    <subjectIdentity>
      <resourceRef
        xlink:href=" http://localhost/SRC/SRC051.aspx "/>
      <resourceRef
        xlink:href=" http://localhost/SRC/SRC052.aspx "/>
    </subjectIdentity>
    <baseName>
      <baseNameString>原始碼051</baseNameString>
    </baseName>
    <baseName>
      <baseNameString>原始碼052</baseNameString>
    </baseName>
  </topic>

```

</topicMap>

上例中，合併前的兩個主題“SRC051”和“SRC052”，在其<subjectIdentity>元素中都具有指向“http://localhost/SRC/SRC051.aspx”的指標，表示它們具有相同的意涵，故合併成為主題“PER18”。



3 系統架構概述

本研究將存在於程式原始碼中關於存取資料的相關資訊擷取出來，除了以 Web 介面呈現及查詢，並增加了 XTM 檔案匯出的功能，讓擷取出來的資料能透過主題地圖進行交換，更進一步地將結果視覺化的方式呈現，具體建構出系統之資料操作藍圖。

本系統架構圖如下圖(圖 3.1)，主要分為程式碼剖析引擎(SCPE)、XTM 檔案匯出(XTMexporter)、及網頁查詢過濾介面(WebViewer)三個模組。本節就系統做一概略性的描述，詳細的模組功能說明，將於下一單元介紹，以下就三個模組進行扼要的描述：

3.1 程式碼剖析引擎模組概述



程式碼剖析引擎模組(SCPE)，將存在於程式原始碼中，有關存取資料，即新增、修改及刪除資料的相關資訊擷取出來，並將相關的資料表及欄位，以及程式碼名稱及位於此程式碼的行數等相關資料，存入資料庫。

3.2 XTM 檔案匯出模組概述

XTM檔案匯出模組(XTMexporter)，提供XTM 檔案匯出機制，將資料庫中相關的資料轉換為XTM 檔案輸出，以利於資料交換，並可匯入其他主題地圖工具中使用。

3.3 網頁查詢過濾介面模組概述

網頁查詢過濾介面模組(WebViewer)，提供網頁查詢過濾的介面，讓使用者可快速地瀏覽及查詢相關的資料，並過濾掉不需要的資料。

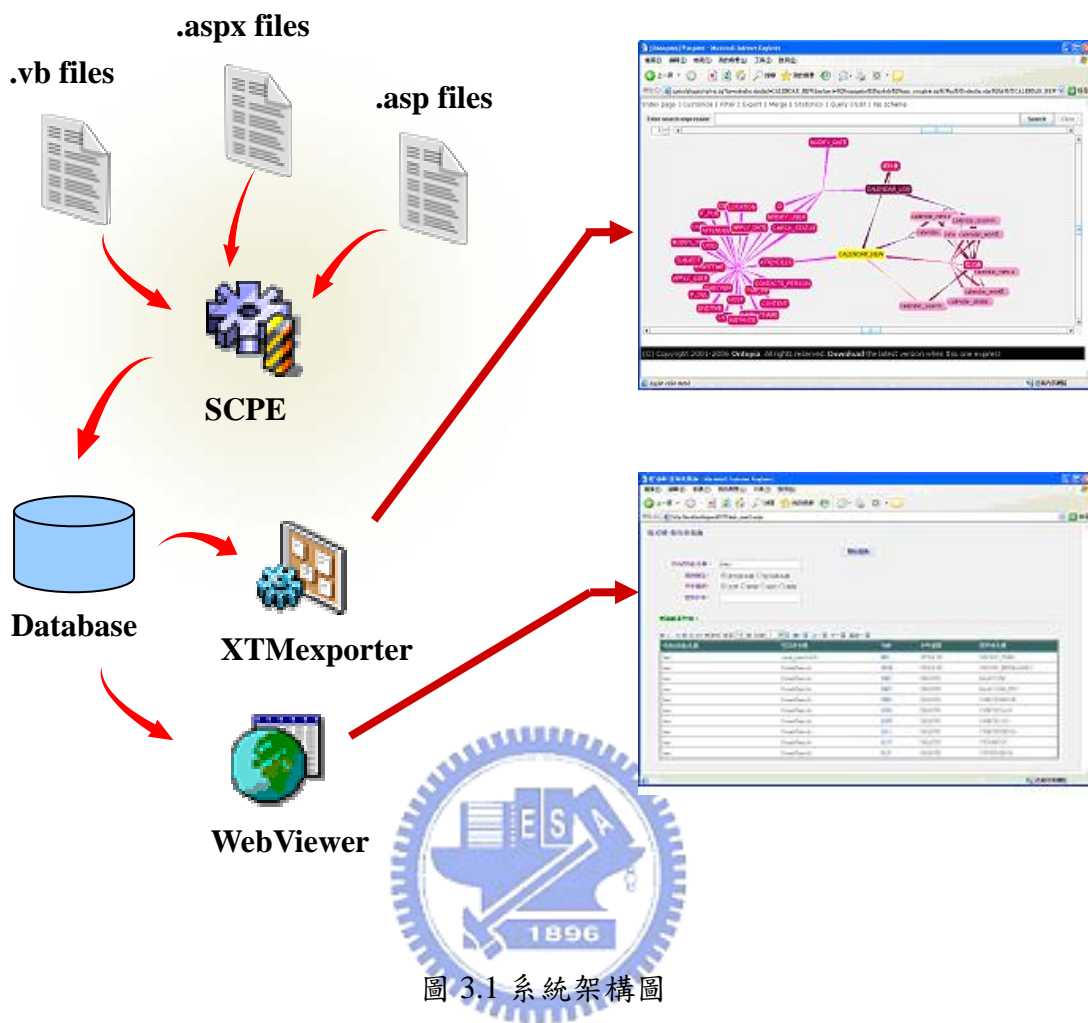


圖 3.1 系統架構圖

4 系統模組說明

此部份將針下各模組做詳細的介紹：

4.1 程式碼剖析引擎模組說明

此模組將存在於程式原始碼中，有關存取資料，即新增(INSERT)、修改(UPDATE)及刪除(DELETE)等相關的資訊擷取出來，並將相關的資料表及欄位，以及程式碼名稱及位於此程式碼的行數等相關資料，存入資料庫。

4.1.1 剖析原始碼中新增資料的資訊(INSERT)

剖析原始碼中新增資料的資訊(INSERT)的流程如下圖 4.1：



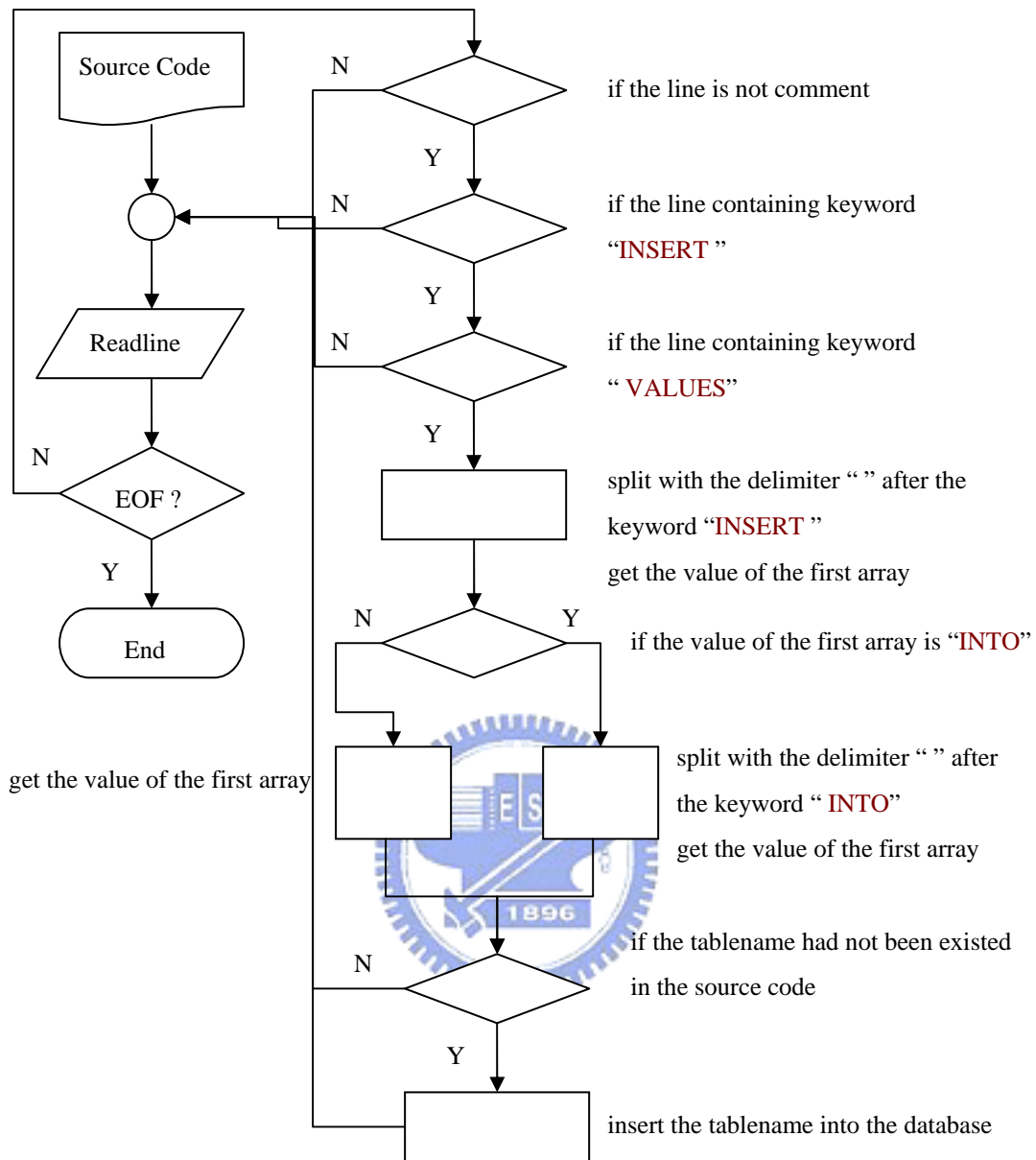


圖 4.1 剖析原始碼中新增資料的資訊之流程圖

以下以實際例子的說明，以下為存在於資料庫應用程式中關於新增資料操作的一行 SQL 敘述：

```

sql="insert into patent (p_number,p_title,p_inventor,p_claim,p_asbstact) values (' &
p_number.text & ',' & p_title.text & ',' & p_inventor.text & ',' & p_claim.text &
',' & p_abstract.text & ')'"
  
```

程式會逐行讀取原始碼檔案直到讀取到檔尾結束(EOF)，首先會判斷此行是

否為註解行(即第一個字元為"");若非註解行,判斷是否有"INSERT"關鍵字;若是,繼續判斷是否有"VALUES"關鍵字;若是,從"INSERT"關鍵字之後以空白字元" "分隔,取出各 token 的值,存入陣列(Array)中,判斷陣列的第一個 token 的值是否為"INTO"關鍵字;若是,從"INTO"關鍵字之後以空白字元" "分隔,取出各 token 的值,存入陣列(Array)中,第一個 token 的值(patent)即為欲得到的資料表名稱;若不是,即為欲得到的資料表名稱(patent)。因程式允許使用者可重複執行,程式最後會判斷資料庫中是否已存在此資訊(patent),若尚未存在,則將此新增的資料表存入資料庫中。

4.1.2 剖析原始碼中修改資料的資訊(UPDATE)

剖析原始碼中修改資料的資訊(UPDATE)的流程如下圖 4.2 :

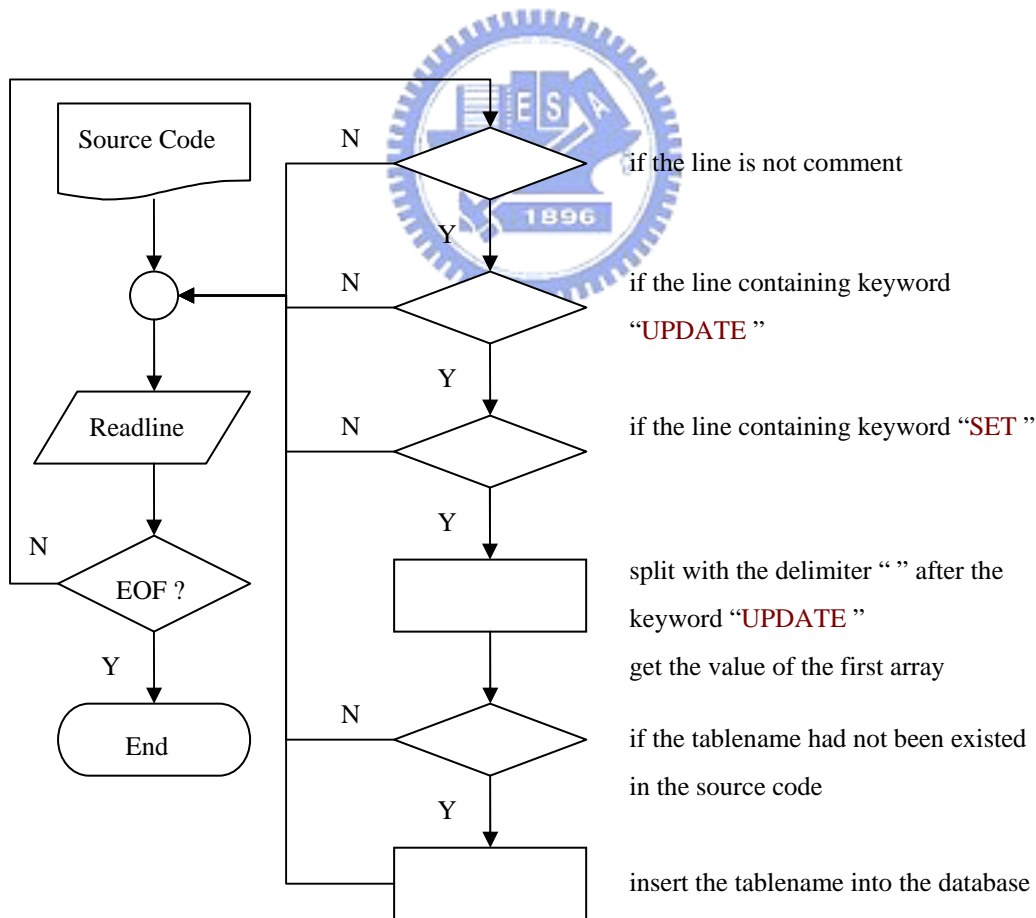


圖 4.2 剖析原始碼中修改資料的資訊之流程圖

以下以實際例子的說明，以下為存在於資料庫應用程式中關於修改資料操作的一行 SQL 敘述：

```
sql = "update group_supplier set s_name='" & TextBoxName.Text & "', s_keyman =  
'" & TextBoxKeyman.Text & "', s_email='" & TextBoxEmail.Text & "',s_tel = '" &  
TextBoxTel.Text & "', s_fax = '" & TextBoxFax.Text & "', s_address= '" &  
TextBoxAddress.Text & "' where s_code='" & TextBoxCode.Text & "'"
```

程式會逐行讀取原始碼檔案直到讀取到檔尾結束(EOF)，首先會判斷此行是否為註解行(即第一個字元為"");若非註解行，判斷是否有"UPDATE"關鍵字；若是，繼續判斷是否有"SET" 關鍵字；若是，從"UPDATE"關鍵字之後以空白字元" "分隔，取出各 token 的值，存入陣列(Array)中，第一個 token 的值 (group_supplier)即為欲得到的資料表名稱；若不是，即為欲得到的資料表名稱 (group_supplier)。因程式允許使用者可重複執行，程式最後會判斷資料庫中是否已存在此資訊(group_supplier)，若尚未存在，則將此修改的資料表存入資料庫中。

4.1.3 剖析原始碼中刪除資料的資訊(DELETE)

剖析原始碼中刪除資料的資訊(UPDATE)的流程如下圖 4.3：

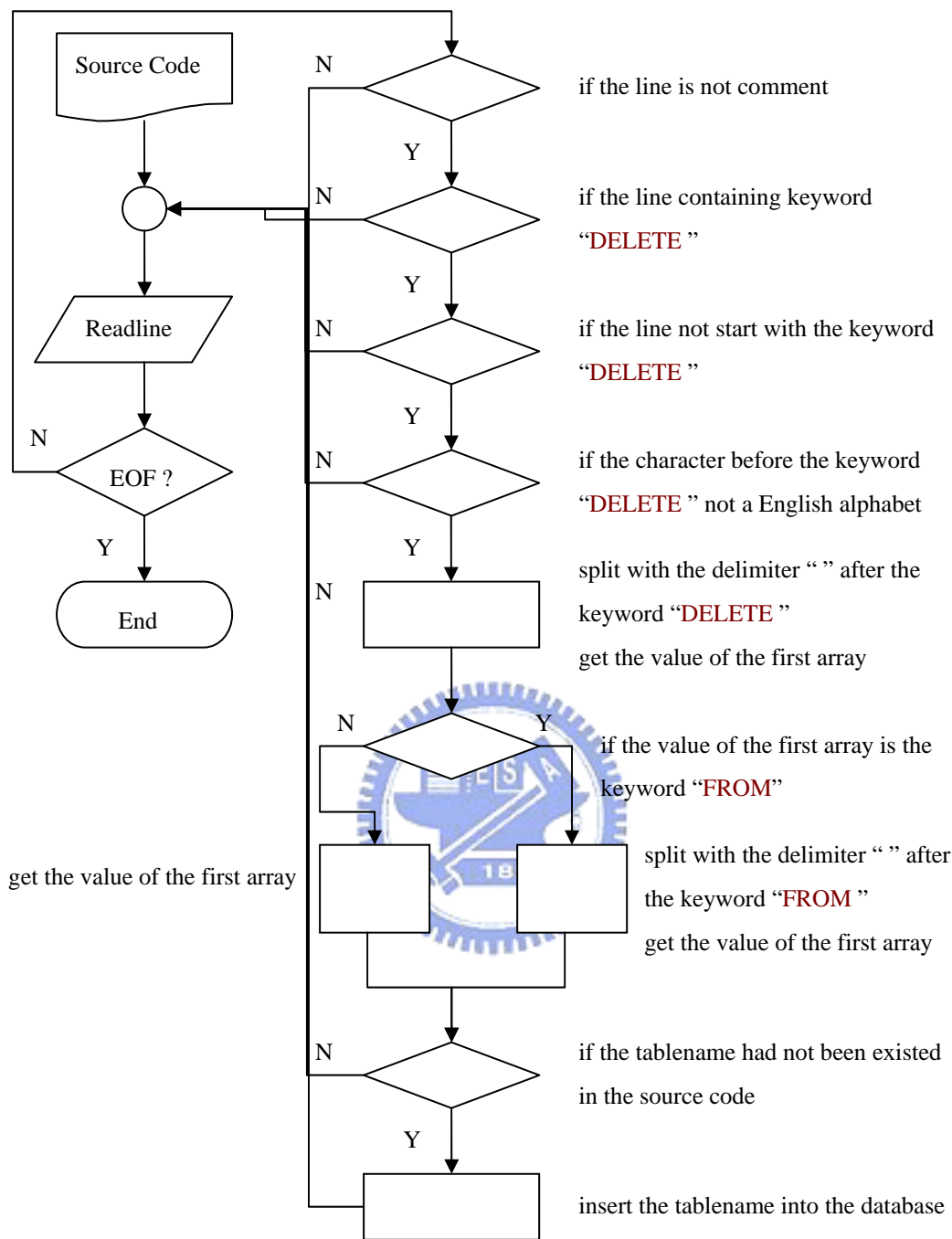


圖 4.3 剖析原始碼中刪除資料的資訊之流程圖

以下以實際例子的說明，以下為存在於資料庫應用程式中關於刪除資料操作的一行 SQL 敘述：

```
sql= "delete from m016 where m016_subjectid = " & subjectid
```

程式會逐行讀取原始碼檔案直到讀取到檔尾結束(EOF)，首先會判斷此行是

否為註解行(即第一個字元為””)；若非註解行，判斷是否有”DELETE”關鍵字；若是，判斷此行是否不是以”DELETE”開始；若是，繼續判斷”DELETE”的前一個字元是否非英文字元；若是，從”DELETE”關鍵字之後以空白字元” “分隔，取出各 token 的值，存入陣列(Array)中，判斷陣列的第一個 token 的值是否為”FROM” 關鍵字；若是，從”FROM”關鍵字之後以空白字元” “分隔，取出各 token 的值，存入陣列(Array)中，第一個 token 的值(m016)即為欲得到的資料表名稱；若不是，即為欲得到的資料表名稱(m016)。因程式允許使用者可重複執行，程式最後會判斷資料庫中是否已存在此資訊(m016)，若尚未存在，則將此刪除的資料表存入資料庫中。

4.1.4 資料庫綱要設計

資料庫綱要(schema)的實體關係圖(Entity Relationship)，如下圖 4.4，有兩個資料表，分別為資料表 PSCT 及資料表 PSCF，資料表 PSCT 存放原始碼中有關資料表相關的資訊，共有 11 個欄位，各欄位說明如下：

psc_id：資料表主鍵值(Primary Key)。

psc_sc：程式原始碼名稱。

psc_type：存取類別。

psc_table：資料表名稱。

psc_line：行數。

psc_dir：原始碼來源路徑。

psc_flag：原始碼來源類型。

psc_sysname：系統(或模組)名稱。

psc_filetyp：原始碼檔案種類。

psc_refData：程式碼說明。

psc_refURL：程式碼說明連結。

PSCF 為存放資料表中相關聯的欄位資訊，各欄位說明如下：

f_id：資料表主鍵值(Primary Key)。

f_fsn：為一外來鍵(Foreign Key)，參考到 PSCT 資料表的 psc_id 欄位。

f_fname：欄位名稱。

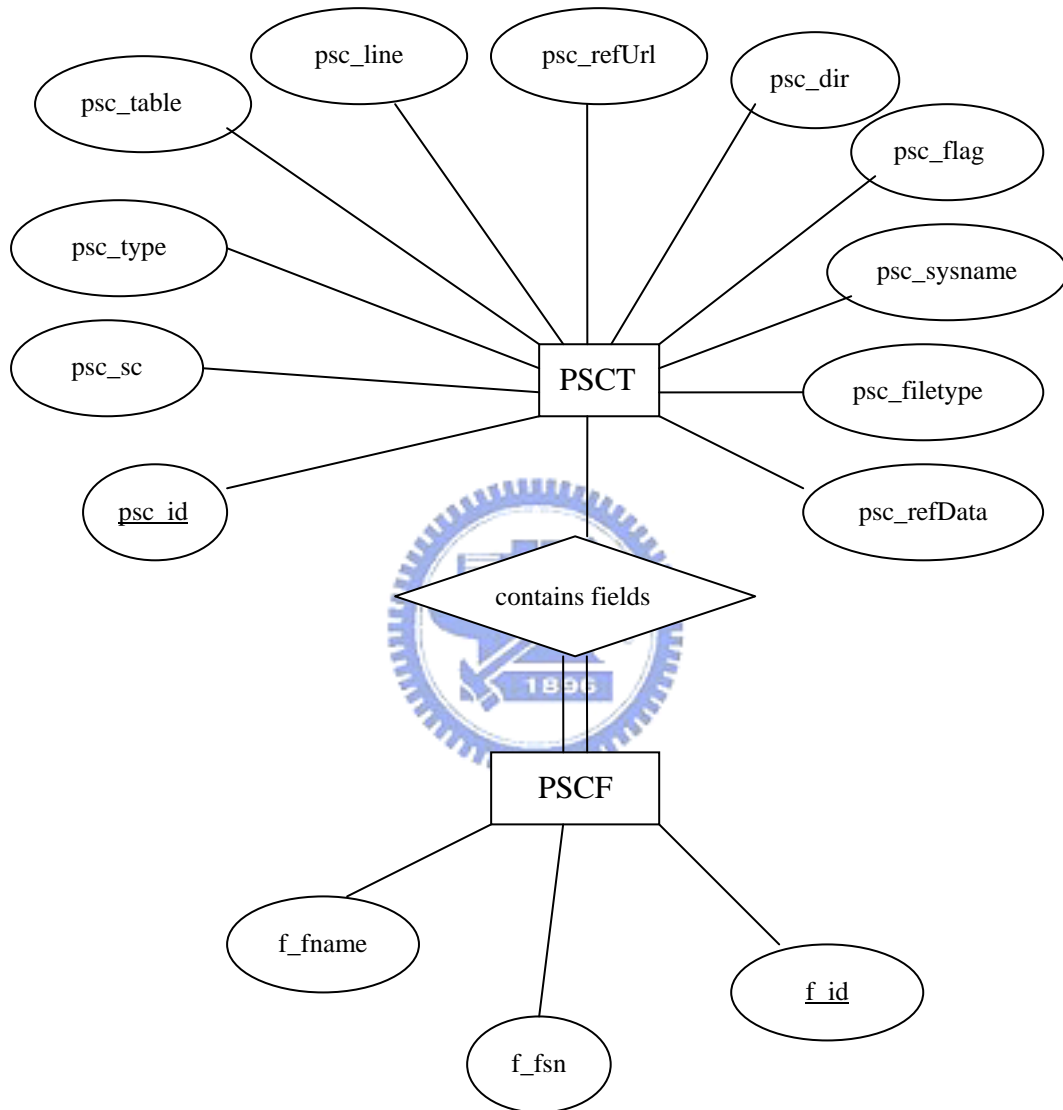


圖 4.4 實體關係圖

4.2 XTM 檔案匯出模組說明

此模組為 XTM 檔案匯出模組，提供 XTM 檔案匯出機制，將資料庫中相關的資料轉換為 XTM 檔案輸出，以利於資料交換，並可匯入其他主題地圖工具中

使用，例如 Ontopia 公司所提供的 Omnigator 知識導覽工具[9]可用來呈現主題地圖的架構。

根據 XTM 的語法，將主題(Topic)、關聯(Association)及資源(Occurrence)相關的資訊匯出，本系統此三種型態的類別如下：

4.2.1 主題類別(Topic Type)

本系統的主題類別包含程式碼及資料表。範例如下：

```
<topic id="SRC002">
  <instanceOf>
    <topicRef xlink:href="#Source" />
  </instanceOf>
  <baseName>
    <baseNameString>原始碼 002</baseNameString>
  </baseName>
</topic>
```

上面範例表示，主題「原始碼 002」的主題類別為「程式碼(Source)」。



4.2.2 關聯類別(Association Type)及關聯角色類別(Association Role Type)

本系統的關聯類別包含新增、修改、刪除及欄位，關聯角色類別包含位於及存在。範例如下：

```
<association>
  <instanceOf>
    <topicRef xlink:href="#UPDATE"/>
  </instanceOf>
  <member>
    <roleSpec>
      <topicRef xlink:href="#locate"/>
    </roleSpec>
    <topicRef xlink:href="#SRC001"/>
  </member>
  <member>
    <roleSpec>
```

```

        <topicRef xlink:href="#exist"/>
    </roleSpec>
    <topicRef xlink:href="#TBL001"/>
</member>
</association>

```

上面範例表示，程式碼 001(SRC001)及資料表 001(TBL001)有一個「UPDATE」關聯類別，且此「UPDATE」關聯類別，有一個關聯角色類別「位於(locate)」表示其位於程式碼 001，另一個關聯角色類別「存在(exsit)」表示其存在於資料表 001。

4.2.3 資源類別(Occurrence Type)

本系統的資源類別包含程式碼說明及程式碼說明連結，範例如下：

```

<topic id="NT0001">
    <instanceOf>
        <topicRef xlink:href="#SRC001" />
    </instanceOf>
    <baseName>
        <baseNameString>原始碼 001 說明</baseNameString>
    </baseName>
    <occurrence>
        <instanceOf>
            <topicRef xlink:href="#Note"/>
        </instanceOf>
        <resourceData>程式碼說明</resourceData>
    </occurrence>
    <occurrence>
        <instanceOf>
            <topicRef xlink:href="#NoteURL"/>
        </instanceOf>
        <resourceRef xlink:href="http://localhost/code/SourceDescription.aspx"/>
    </occurrence>
</topic>

```

上面範例表示，「原始碼 001 說明(NT0001)」有一個主題類別「原始碼 001(SRC001)」，其有兩個資源來源，一為文字說明(resourceData)「程式碼

說明」，另一個為參考的連結(resourceRef)

「<http://localhost/code/SourceDescription.aspx>」。

4.3 網頁查詢過濾介面模組說明

此模組提供網頁查詢過濾的介面，讓使用者可快速地瀏覽及查詢相關的資料，並過濾掉不需要的資料。查詢的項目如下：

1. 系統(模組)名稱：可依整個系統或系統的某一個模組查詢。
2. 存取種類：可依新增、修改、刪除的存取方式或全部查詢。
3. 查詢類型：分為資料表名稱及程式碼名稱查詢。
4. 查詢字串：依查詢類型進行關鍵字之查詢。



5 系統實作與呈現

本章描述系統的實作及結果的呈現。第一個部分將說明系統開發平台及系統開發語言與工具，第二部份是程式碼剖析引擎的實作說明，第三部份則是XTM檔案匯出的實作，第四部份為網頁查詢過濾介面的實作，第五部份以視覺化主題地圖方式呈現結果，第六部份以Web介面呈現及查詢。

5.1 開發平台及工具

本系統的開發平台及開發語言與工具，如下表5.1所示：

系統開發平台	Operating System	Microsoft Windows XP Professional
	Application Server	Internet Information Service 5.1
	Database Server	Microsoft SQL Server 2005
系統開發工具	.Net Framework 2.0 MDAC 2.8	
系統開發語言	Visual Basic .Net	
程式剖析開發環境	Microsoft Visual Studio 2005	
Web 瀏覽開發環境	Microsoft Visual Studio .Net	

表 5.1：系統開發平台及開發語言與工具

5.2 程式碼簡述及選取

實作的程式原始碼來源如下表 5.2，有三種不同程式語言類型(asp、aspx 及 vb)，共 10 個資料庫應用程式，總共 446 個原始碼檔案。

資料庫應用程式名稱	程式語言類型	原始碼檔案數
儲存媒體管理系統	vb	69
生產紀錄管理資訊系統	vb	147
問卷調查	asp	27

線上報名	asp	42
討論區	asp	39
意見箱	aspx	15
留言板	aspx	25
行事曆	aspx	25
公佈欄	aspx	26
工作管理	aspx	31

表 5.2：實作的程式原始碼列表

以下為此三種不同類型的原始碼格式，以下節錄此三種類型原始碼與存取資料庫相關的部份，分別如下：

5.2.1 asp 檔案格式



```

<%
Set RSreg = Server.CreateObject("ADODB.RecordSet")
subject = replace(trim(request("subject")), "", "")
subjectid = request("subjectid")
if subjectid <> "" then
    sql = "" & _
        " delete from m011 where m011_subjectid = " & subjectid & "; " & _
        " delete from m012 where m012_subjectid = " & subjectid & "; " & _
        " delete from m013 where m013_subjectid = " & subjectid & "; " & _
        " delete from m014 where m014_subjectid = " & subjectid & "; " & _
        " delete from m015 where m015_subjectid = " & subjectid & "; " & _
        " delete from m016 where m016_subjectid = " & subjectid
    conn.execute(sql)
    response.end
end if
notetype_str = trim(request("notetype"))
notetype = ""
if ( subjectid <> "" then
    sql = "" & _

```



```

" update m011 set " & _
" m011_subject = "" & subject & "", " & _
" m011_questno = " & request("questno") & ", " & _
" m011_bdate = "" & bdate & "", " & _
" m011_edate = "" & edate & "", " & _
" m011_online = "" & request("online") & "", " & _
" m011_notetype = "" & notetype & "", " & _
" m011_questionnote= "" & replace(trim(request("questionnote")), "", "") & "", " & _
" m011_haveprize = "" & request("haveprize") & "", " & _
" m011_jumpquestion = "" & request("jumpquestion") & "", " & _
" m011_onlyonce = "" & request("onlyonce") & "", " & _
" m011_modifyuser = "" & session("NetUser") & "", " & _
" m011_updatetime = getdate() " & _
" where " & _
" m011_subjectid = " & subjectid
conn.execute(sql)
response.end
end if
set rs = conn.execute("select IsNull(Max(m011_subjectid), 0) from m011")
subjectid = rs(0) + 1
sql = "" & _
" insert into m011 ( " & _
" m011_subjectid, " & _
" m011_subject, " & _
" m011_questno, " & _
" m011_bdate, " & _
" m011_edate, " & _
" m011_online, " & _
" m011_notetype, " & _
" m011_questionnote, " & _
" m011_haveprize, " & _
" m011_jumpquestion, " & _
" m011_onlyonce, " & _
" m011_createuser, " & _
" m011_createtime, " & _
" m011_modifyuser, " & _
" m011_updatetime " & _
" ) values ( " & _

```



```

subjectid & ", " & _
" " & subject & ", " & _
request("questno") & ", " & _
" " & bdate & ", " & _
" " & edate & ", " & _
" " & request("online") & ", " & _
" " & notetype & ", " & _
" " & Replace(trim(request("questionnote")), "", "") & ", " & _
" " & request("haveprize") & ", " & _
" " & request("jumpquestion") & ", " & _
" " & request("onlyonce") & ", " & _
" " & session("NetUser") & ", " & _
" getdate(), " & _
" " & session("NetUser") & ", "& _
" getdate() " & _
" )"
conn.execute(sql)
response.redirect "02_add1.asp?subjectid=" & subjectid
%>

```



5.2.2 aspx 檔案格式

```

<Script Language="VB" Runat="Server">
    Sub Page_Load(sender As Object, e As EventArgs)
        Dim sql="select max(p_number) from patent"
        dim ds as new dataset()
        select Case request("db")
            case "mssql"
                Dim cn As New
SqlConnection(ConfigurationSettings.AppSettings("ConnString_mssql"))
                dim da as sqlDataAdapter =new sqlDataAdapter(sql,cn)
                da.fill(ds,"patent")
            case "access"
                Dim cn As New
OleDbConnection(ConfigurationSettings.AppSettings("ConnString_access"))
                dim da as OleDbDataAdapter =new OleDbDataAdapter(sql,cn)
                da.fill(ds,"patent")
        'cn.close
    End Sub

```

```

        case "excel"
            Dim cn As New
OdbcConnection(ConfigurationSettings.AppSettings("ConnString_excel"))
            dim da as OdbcDataAdapter =new OdbcDataAdapter(sql,cn)
            da.fill(ds,"patent")
        end select
        if isDBNull(ds.tables("patent").rows(0).item(0)) then
            p_number.text=(Year(Today)-1911)*10000 + 1
        Else
            p_number.text=ds.tables("patent").rows(0).item(0) + 1
        End if
    End Sub
    Sub btnOk_Click(sender As Object, e As EventArgs)
        Dim sql as String
        sql="insert into patent values ('" & p_number.text & "','" & p_title.text & "','" &
p_inventor.text & "','" & p_claim.text & "','" & p_abstract.text & "')"
        select Case request("db")
            case "mssql"
                Dim cn As New
SqlConnection(ConfigurationSettings.AppSettings("ConnString_mssql"))
                Dim cmd as new sqlcommand(sql,cn)
                cn.open
                cmd.ExecuteNonQuery()
                cn.close
            case "access"
                Dim cn As New
OleDbConnection(ConfigurationSettings.AppSettings("ConnString_access"))
                Dim cmd as new OleDbcommand(sql,cn)
                cn.open
                cmd.ExecuteNonQuery()
                cn.close
            case "excel"
                Dim cn As New
OdbcConnection(ConfigurationSettings.AppSettings("ConnString_excel"))
                Dim cmd as new Odbccommand(sql,cn)
                cn.open
                cmd.ExecuteNonQuery()
                cn.close

```

```

end select
        response.redirect("mssql_list.aspx?db=" & request("db"))
End Sub
</Script>
<title>mssql_new</title>
<LINK href="css.css" type="text/css" rel="stylesheet">
<form id="Form1" method="post" runat="server" >
    <div align="right"><a href="#" onClick="history.go(-1);">回上一頁</a></div>
    <table width="90%" border="1" align="center">
        <caption><font color="blue">「<%=request("db")%>」新增資料
</font></caption>
        <tr>
            <td width="15%">專利號碼：</td>
            <td><asp:label id="p_number" runat="server" Width="100px"
></asp:label></td>
        </tr>
        <tr>
            <td>發明人：</td>
            <td><asp:textbox id="p_inventor" runat="server"
Width="300px"></asp:textbox></td>
        </tr>
        <tr>
            <td>發明名稱：</td>
            <td><asp:textbox id="p_title" runat="server" Width="600px"
></asp:textbox></td>
        </tr>
        <tr>
            <td>專利權人：</td>
            <td><asp:textbox id="p_claim" runat="server" Width="600px"
></asp:textbox></td>
        </tr>
        <tr>
            <td>摘要：</td>
            <td><asp:textbox id="p_abstract" runat="server" Width="600px"
Height="200px" TextMode="MultiLine"></asp:textbox></td>
        </tr>
    </table>
<br/>

```



```

        <div align="center"><asp:button id="btnOk" runat="server" Width="40px"
Text="確定" CssClass="sbtttn" onClick="btnOk_Click"></asp:button><input type="reset"
id="btn_reset" name="btn_reset" class="sbtttn" value="重設" /></div>
    </form>

```

5.2.3 vb 檔案格式

```
Public Class group_supplier
```

```
    Dim cn As SqlConnection
```

```
    Dim ds As New DataSet()
```

```
    Private Function checkData() As Boolean
```

```
        Dim sc As String
```

```
        sc = "select * from group_supplier where s_code = '" & TextBoxCode.Text & "'"
```

```
        Dim cmd As SqlCommand = New SqlCommand(sc, cn)
```

```
        Dim dr As SqlDataReader = cmd.ExecuteReader
```

```
        If dr.Read Then
```

```
            checkData = False
```

```
        Else
```

```
            checkData = True
```

```
        End If
```

```
        dr.Close()
```

```
    End Function
```

```
    Private Sub bindData()
```

```
        Dim sc As String
```

```
        sc = "select * from group_supplier"
```

```
        Dim da As SqlDataAdapter = New SqlDataAdapter(sc, cn)
```

```
        da.Fill(ds, "group_role")
```

```
        If ds.Tables("group_role").Rows.Count > 0 Then
```

```
            DataGrid1.DataSource = ds
```

```
            DataGrid1.DataMember = "group_role"
```

```
            bm = Me.BindingContext(ds, "group_role")
```

```
            ButtonModify.Enabled = True
```

```
            ButtonDel.Enabled = True
```

```
            ButtonRefresh.Enabled = True
```

```
            ButtonPrint.Enabled = True
```

```
        Else
```

```
            ButtonModify.Enabled = False
```

```
            ButtonDel.Enabled = False
```



```

        ButtonRefresh.Enabled = False
        ButtonPrint.Enabled = False
    End If
End Sub
Private Sub ButtonAdd_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles ButtonAdd.Click
    If checkInput() Then
        cn.Open()
        If checkData() Then
            Dim ic As String
            ic = "insert into group_supplier values (" & TextBoxCode.Text & "," &
            TextBoxName.Text & "," & TextBoxKeyman.Text & "," & TextBoxTel.Text & "," &
            TextBoxFax.Text & "," & TextBoxAddress.Text & "," & TextBoxEmail.Text & ")"
            Dim cmd As SqlCommand = New SqlCommand(ic, cn)
            cmd.ExecuteNonQuery()
            log.InsertLog(ic, Me.Name, cn)
            clearDataBindings()
            bindData()
            bm.Position = bm.Count - 1
            MsgBox("新增資料成功!", MsgBoxStyle.Information)
        Else
            MsgBox("此筆資料已存在!", MsgBoxStyle.Critical)
        End If
        cn.Close()
    End If
End Sub
Private Sub ButtonModify_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles ButtonModify.Click
    If checkInput() Then
        cn.Open()
        If Not checkData() Then
            Dim uc As String
            uc = "update group_supplier set s_name=" & TextBoxName.Text & ", s_keyman
            = " & TextBoxKeyman.Text & ", s_email=" & TextBoxEmail.Text & ",s_tel = " & TextBoxTel.Text
            & ", s_fax = " & TextBoxFax.Text & ", s_address=" & TextBoxAddress.Text & " where
            s_code=" & TextBoxCode.Text & ""
            Dim cmd As SqlCommand = New SqlCommand(uc, cn)
            cmd.ExecuteNonQuery()

```

```

        ds.AcceptChanges()
        log.InsertLog(uc, Me.Name, cn)
        MsgBox("修改資料成功！", MsgBoxStyle.Information)
    Else
        MsgBox("此筆資料不存在！", MsgBoxStyle.Critical)
    End If
    cn.Close()
End If
End Sub
Private Sub ButtonDel_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles ButtonDel.Click
    cn.Open()
    If Not checkData() Then
        If MsgBox("是否刪除此筆資料?", MsgBoxStyle.YesNo) = MsgBoxResult.Yes Then
            Dim dc As String
            dc = "delete from group_supplier where s_code=" & TextBoxCode.Text & ""
            Dim cmd As SqlCommand = New SqlCommand(dc, cn)
            cmd.ExecuteNonQuery()
            ds.AcceptChanges()
            bm.RemoveAt(bm.Position)
            log.InsertLog(dc, Me.Name, cn)
        End If
    Else
        MsgBox("此筆資料不存在！！", MsgBoxStyle.Critical)
    End If
    cn.Close()
End Sub
Private Function getRptNum(ByVal csName As String) As Integer
    Dim sc As String
    sc = "select count(*) as 'RptNum' from rptnum_mag where rptnum_cserial=" & csName & ""
and rptnum_date = " & Today & ""
    Dim cmd As SqlCommand = New SqlCommand(sc, cn)
    Dim dr As SqlDataReader = cmd.ExecuteReader
    If dr.Read Then
        getRptNum = dr.Item("RptNum")
    Else
        getRptNum = 0
    End If
End Function

```

```

dr.Close()
End Function
Private Sub insertRptTable(ByVal ReportCode As String)
    Dim nInOut As Integer
    nInOut = getRptNum(ReportCode) + 1
    Dim ic As String
    ic = "insert into rptnum_mag values ('" & ReportCode & "','" & Today & "','" &
String.Format("{0:000}", nInOut) & "')"
    Dim icmd2 As SqlCommand = New SqlCommand(ic, cn)
    icmd2.ExecuteNonQuery()
    ReportNo = ReportCode & Year(Today) & String.Format("{0:00}", Month(Today)) &
String.Format("{0:00}", Microsoft.VisualBasic.DateAndTime.Day(Today)) &
String.Format("{0:000}", nInOut)
End Sub

```

5.3 資料庫綱要實作

根據 4.1.4 的資料庫綱要(schema)設計的實體關係圖(ER-Model)，將其繪製為以下的實作資料表關聯圖(Rational-Model)，如下圖 5.1：

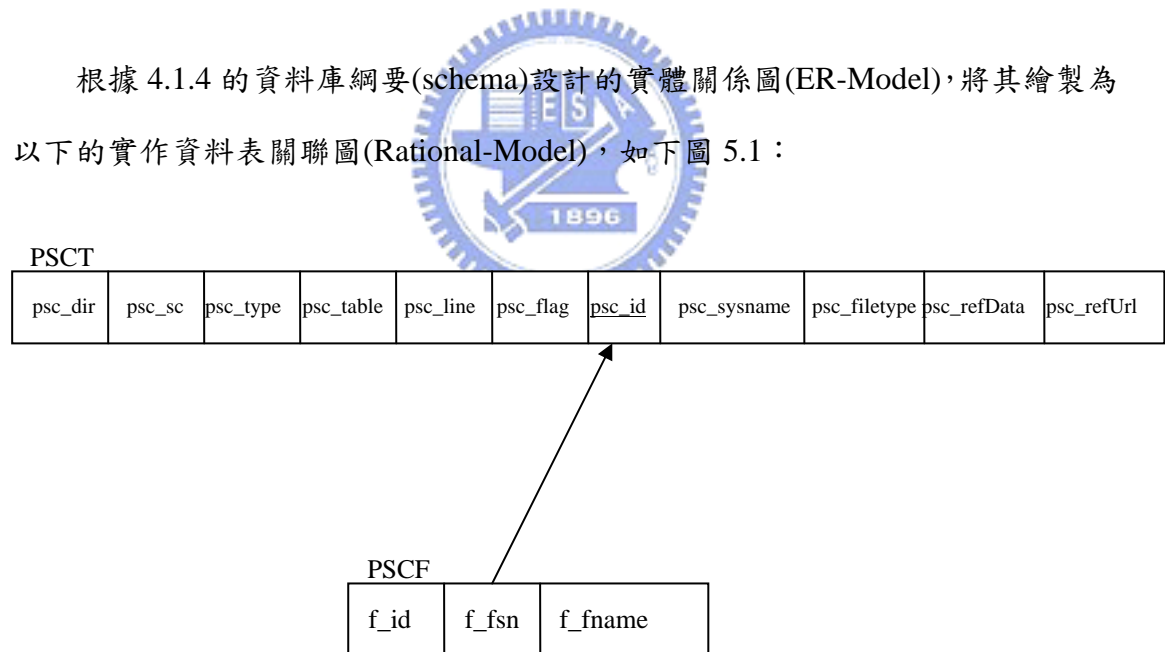


圖 5.1 資料表關聯圖

5.4 程式原始碼剖析

5.4.1 使用者介面

程式碼剖析的使用者介面設計如下圖 5.2，系統設計為可允許剖析多個系統(或模組)，剖析的程式原始碼。程式碼的檔案種類，可於資料表中設定，本研究的資料來源主要以 asp、aspx 及 vb 三種檔案類型的資料為主，對於其他程式語言(如 C++、JAVA、PHP...等)，可於資料表(CONJ)中，設定其相關的程式語法即可進行剖析，資料表(CONJ)設定如下圖 5.3。程式碼來源，可依使用者選擇剖析某個檔案或某個目錄下的檔案。若程式碼存在於子目錄中，可勾選「包含子目錄」，將一併以下所有子目錄下的所有檔案進行剖析。

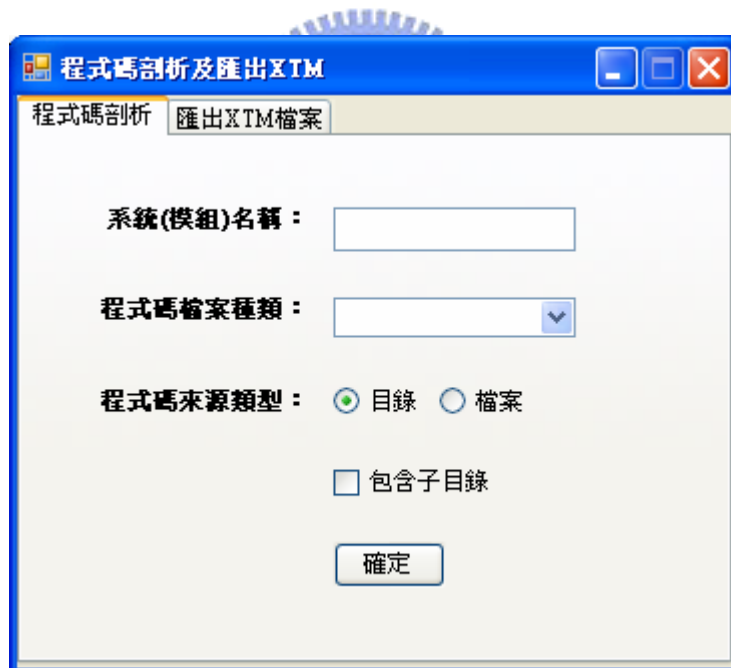


圖 5.2 程式碼剖析使用者介面

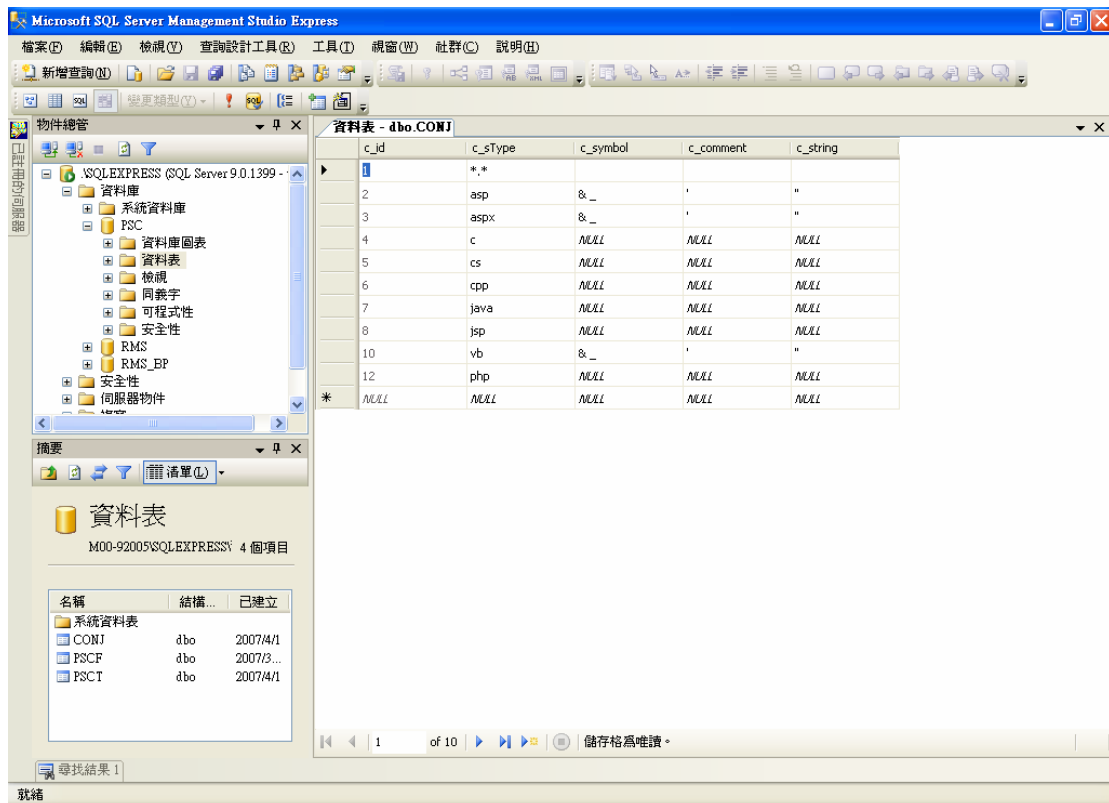


圖 5.3 各程式語法設定資料表

c_sType：程式原始碼檔案之附檔名。

c_symbol：該程式語法多行連結符號。

c_comment：該程式語法註解的符號。

c_string：該程式語法字串連結符號。

5.4.2 程式演算法

依 4.1 設計的程式流程圖，實作剖析的程式演算法如下：

5.4.2.1 剖析原始碼中新增資料的資訊(INSERT)

剖析原始碼中新增資料的資訊(INSERT)的 pseudocode 如下，程式以逐行讀取原始碼檔案的方式進行，首先判斷此行不是註解，判斷含有”INSERT”資料新增操作的關鍵字，再判斷含有”VALUES”關鍵字，從”INSERT”後，以空白字元為分隔，判斷第一個陣列的值是否為”INTO”，若不是，此第一個陣列的值，即為

資料表的名稱；若是，則從”INTO”後，以空白字元分隔，取出第一個陣列的值，過濾”(”字元，即為所要的資料表名稱。因程式允許重複執行，判斷若資料庫尚未存在此資料表資訊，則存入此資訊。之後剖析此資料表的相關欄位，從”INSERT”關鍵字後，以”)”字元分隔，判斷第一個陣列值含有”VALUES”關鍵字，以”(”字元分隔，取出第二個陣列之值並以”,”分隔，即得到此資料表相關欄位的陣列，同樣的判斷若資料庫尚未存在此欄位資訊，則存入此資訊。

Pseudocode :

if the line is not comment

 if the line containing keyword “INSERT ”

 if the line containing keyword “VALUES”

 split with the delimiter “ ” after the keyword “INSERT ”

 if the value of the first array is “INTO”

 split with the delimiter “ ” after the keyword “ INTO”

 get the value of the first array

 if it containing the character “(”

 split with delimiter “(”

 get the value of the first array

 end if

 else

 get the value of the first array

 end if

 if the tablename had not been existed in the source code

 insert the tablename into the database

 split with delimiter “)” after the keyword “INSERT ”

 if the value of the first array not containing keyword “VALUES”

 split it with delimiter “(”

 split the value of the second array with delimiter “,”

 if the fields had not been existed in this tablename

 insert the fields into the database

 end if

 end if

 end if

 end if

end if

end if

5.4.2.2 剖析原始碼中修改資料的資訊(UPDATE)

剖析原始碼中修改資料的資訊(UPDATE)的 pseudocode 如下，程式以逐行讀取原始碼檔案的方式進行，首先判斷此行不是註解，判斷含有” UPDATE”資料修改操作的關鍵字，再判斷含有”SET”關鍵字，從” UPDATE”後，以空白字元為分隔，取出第一個陣列的值，過濾”(”字元，即為資料表的名稱，因程式允許重複執行，判斷若資料庫尚未存在此資料表資訊，則存入此資訊。之後剖析此資料表的相關欄位，取出從”SET”關鍵字之後的字串，判斷是否含有”WHERE”關鍵字，若有則取出”SET”與”WHERE”之間的字串，以”,”字元分隔後，再以”=”分隔，即得到此資料表相關欄位的陣列，同樣的判斷若資料庫尚未存在此欄位資訊，則存入此資訊。



Pseudocode :

if the line is not comment

 if the line containing keyword “UPDATE”

 if the line containing keyword “SET”

 split with the delimiter “ ” after the keyword “UPDATE”

 get the value of the first array

 if it containing the character “(”

 split with delimiter “(”

 get the value of the first array

 end if

 if the tablename had not been existed in the source code

 insert the tablename into the database

 get the string after the keyword “SET”

 if the former string containing the keyword “WHERE”

 get the substring between “SET” and “WHERE”

 end if

 split the result string with delimiter “,”

 if the subarray containing the character “=”

 split the subarray with delimiter “=”

```

        if the fields had not been existed in this tablename
            insert the fields into the database
        end if
    end if
end if
end if
end if
end if
end if
end if

```

5.4.2.3 剖析原始碼中刪除資料的資訊(DELETE)

剖析原始碼中刪除資料的資訊(DELETE)的 pseudocode 如下，程式以逐行讀取原始碼檔案的方式進行，首先判斷此行不是註解，判斷含有”DELETE”資料刪除操作的關鍵字，再判斷不是以”DELETE”開頭，判斷”DELETE”關鍵字前一個字元不屬於英文字元，然後從”DELETE”後，以空白字元為分隔，判斷第一個陣列的值是否為”FROM”，若不是，此第一個陣列的值，即為資料表的名稱；若是，則從”FROM”後，以空白字元分隔，取出第一個陣列的值，過濾”(”及””字元，即為所要的資料表名稱。因程式允許重複執行，判斷若資料庫尚未存在此欄位資訊，則存入此資訊。

Pseudocode :

```

if the line is not comment
    if the line containing keyword “DELETE ”
        if the line not start with the keyword “DELETE ”
            if the character before the keyword “DELETE ” not a English alphabet
                split with the delimiter “ ” after the keyword “DELETE ”
                get the value of the first array
                if the value of the first array is the keyword “FROM”
                    split with the delimiter “ ” after the keyword “FROM ”
                    get the value of the first array
                else
                    get the value of the first array
                end if
            end if
        end if
    end if
end if

```

```
    if the acquired value containing the character "("
        get the value before the character "("
    end if
    if the acquired value containing the character """"
        get the value before the character """"
    end if
    if the tablename had not been existed in the source code
        insert the tablename into the database
    end if
end if
end if
end if
```

5.5 XTM 檔案匯出

由於程式原始碼剖析引擎擷取出的資訊是存入資料庫，為了呈現視覺化的主題地圖，需將相關的資訊轉換為 XTM 的格式，以利資料交換的便利性，所以加入了 XTM 檔案匯出的功能。



5.5.1 使用者介面

XTM 檔案匯出的使用者介面如下圖 5.4，使用者依輸入的系統(或模組)名稱，匯出相關的主題(Topic)、關聯(Association)及資源(Occurrence)等資訊，匯出的 XTM 格式，可參考附錄 C：XTM 檔案範例。

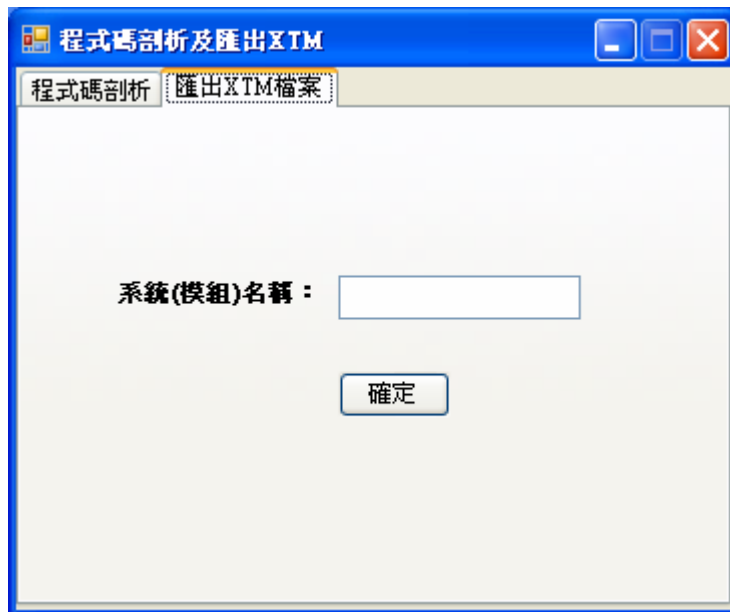


圖 5.4 XTM 匯出之使用者介面

5.5.2 XTM 檔案格式

本系統主要為探討資料庫應用程式之程式碼與資料表之間，依據 XTM 的語法，定義了兩個主題類別程式碼及資料表，對應到每個實體的程式碼及資料表主題，定義新增(INSERT)、修改(UPDATE)及刪除(DELETE)等三個關聯類別，分別代表實體的程式碼與資料表之間的關係為新增、修改及刪除，另定義了欄位關聯類別，表示實體資料表與其欄位之間的關係。若程式原始碼第一行有定義相關的參考資源，例如 RefData~表示有程式碼的說明文字，RefURL~則表示參考到一個程式碼說明的連結網頁。以下摘錄部份的 XTM 檔案說明，詳細的 XTM 範例如附錄 C 所示。

5.5.2.1 定義主題類別(Topic Type)

```
<topic id="Source">  
  <baseName>  
    <baseNameString>程式碼</baseNameString>  
  </baseName>  
</topic>
```

```
<topic id="Table">
  <baseName>
    <baseNameString>資料表</baseNameString>
  </baseName>
</topic>
```

上例中定義兩個主題類別，分別為「程式碼」及「資料表」，分別表示主題「原始碼」及主題「資料表」的類別。

5.5.2.2 定義主題(Topic)

```
<topic id="calendar_new.aspx.vb">
  <instanceOf>
    <topicRef xlink:href="#Source"/>
  </instanceOf>
  <baseName>
    <baseNameString>calendar_new.aspx.vb</baseNameString>
  </baseName>
</topic>
```

上例中表示主題「原始碼(calendar_new.aspx.vb)」的主題類別為「程式碼 (Source)」。



5.5.2.3 定義關聯類別(Association Type)

```
<topic id="INSERT">
  <baseName>
    <baseNameString>新增</baseNameString>
  </baseName>
</topic>
<topic id="UPDATE">
  <baseName>
    <baseNameString>修改</baseNameString>
  </baseName>
</topic>
<topic id="DELETE">
  <baseName>
    <baseNameString>刪除</baseNameString>
```



```

    </baseName>
  </topic>
  <topic id="Field">
    <baseName>
      <baseNameString>欄位</baseNameString>
    </baseName>
  </topic>

```

上例中定義三個關聯類別，分別為「新增」、「修改」及「刪除」。分別表示主題「程式碼」與主題「資料表」之間的關聯。另一個關聯類別-欄位，表示主題「資料表」與主題「欄位」之間的關聯。

5.5.2.4 定義關聯角色類別(Association Role Type)

```

<topic id="locate">
  <baseName>
    <baseNameString>位於</baseNameString>
  </baseName>
</topic>
<topic id="exist">
  <baseName>
    <baseNameString>存在</baseNameString>
  </baseName>
</topic>

```



上例中定義了兩個關聯角色類別，分別為「位於」與「存在」。分別表示欄位「位於(locate)」程式碼及「存在(exist)」於資料表。

5.5.2.5 定義關聯 (Association)

```

<association>
  <instanceOf>
    <topicRef xlink:href="#DELETE"/>
  </instanceOf>
  <member>
    <roleSpec>
      <topicRef xlink:href="#locate"/>
    </roleSpec>
  </member>
</association>

```

```

    <topicRef xlink:href="#calendar_search_list.aspx.vb"/>
</member>
<member>
  <roleSpec>
    <topicRef xlink:href="#exist"/>
  </roleSpec>
  <topicRef xlink:href="#CALENDAR_NEW"/>
</member>
</association>

```

上例中表示關聯類別為「新增」(INSERT)，其有兩個成員，第一個成員的角色為「位於」(locate)，參考到的主題是「原始碼(calendar_search_list.aspx.vb)」一；第二個成員的角色為「存在」(exist)，參考到的主題是「資料表(CALENDAR_NEW)」。

5.5.2.6 定義資源類別(Occurrence Type)

```

<topic id="Note">
  <baseName>
    <baseNameString>程式碼說明</baseNameString>
  </baseName>
</topic>
<topic id="NoteURL">
  <baseName>
    <baseNameString>程式碼說明連結</baseNameString>
  </baseName>
</topic>

```

上例中定義了兩個資源類別，為「程式碼說明」及「程式碼說明連結」。分別表示「程式碼說明」及「程式碼說明連結」的類別。

5.5.2.7 定義資源 (Occurrence)

```

<topic id="NT_mssql_modify.aspx">
  <instanceOf>
    <topicRef xlink:href="#mssql_modify.aspx"/>
  </instanceOf>

```

```

<baseName>
  <baseNameString>mssql_modify.aspx(程式碼說明)</baseNameString>
</baseName>
<occurrence>
  <instanceOf>
    <topicRef xlink:href="#NoteURL"/>
  </instanceOf>
  <resourceRef xlink:href="http://localhost/SourceCodeDescription.aspx"/>
</occurrence>
</topic>

```

若程式碼中有定義程式碼說明標籤(RefData~)及程式碼說明連結標籤(RefURL~)，則會產生如上例中資源的相關標籤，上例中表示「資源(NT_mssql_modify.aspx)」參考到主題「原始碼(mssql_modify.aspx)」，有一個「mssql_modify.aspx(程式碼說明)」參考到「外部參考資源(NoteURL)」，即一個外部的連結(<http://localhost/SourceCodeDescription.aspx>)。

5.6 視覺化主題地圖呈現

在4.2提到Ontopia 公司所提供的Omnigator 知識導覽工具，可用來呈現主題地圖的架構。以下為將程式碼剖析引擎匯出的XTM檔案，匯入Omnigator之後所呈現的結果，以「專利(patent)」資料庫應用程式為例，說明如下：

下圖5.5為載入專利XTM檔案之後進入的索引頁，XTM檔案匯出模組，依據5.5.2定義的主題、關聯及資源，列出的主題類別、關聯類別、關聯成員角色種類、資源指引類別。

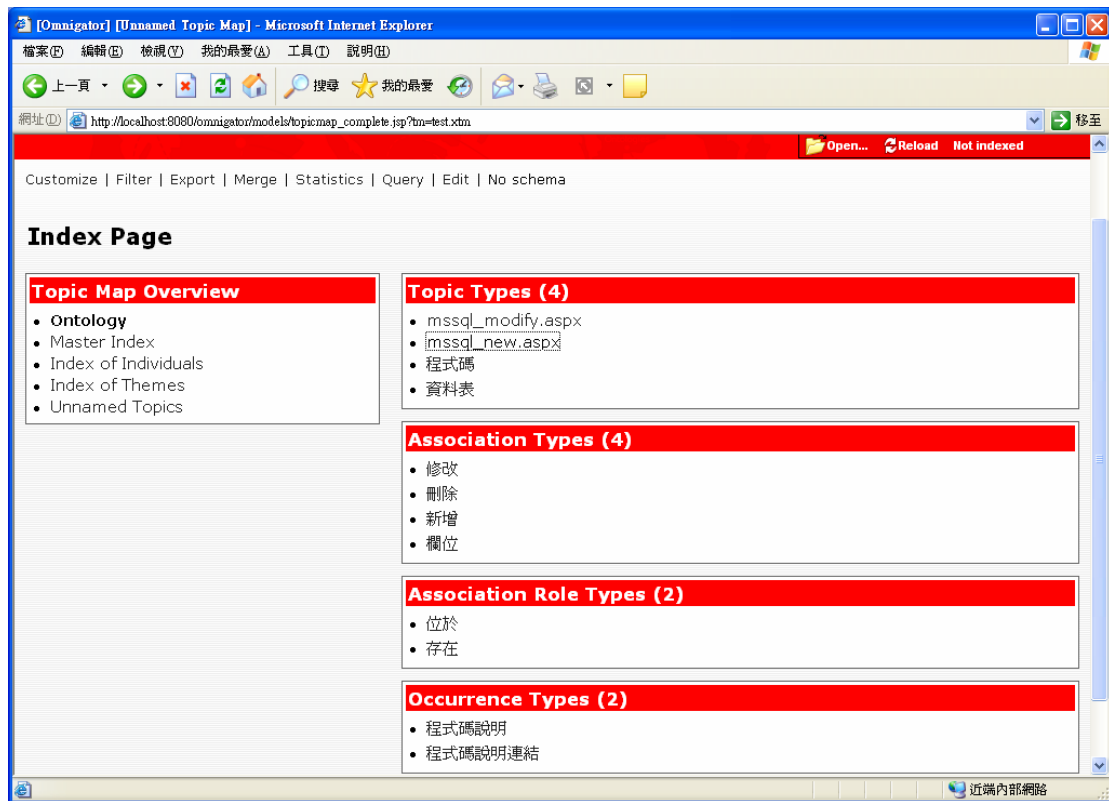


圖 5.5 索引頁

進入每個類別之後，會顯示此類別中所有的主題，例如進入「程式碼」主題類別後，共列出3個程式原始碼，如下圖5-6。

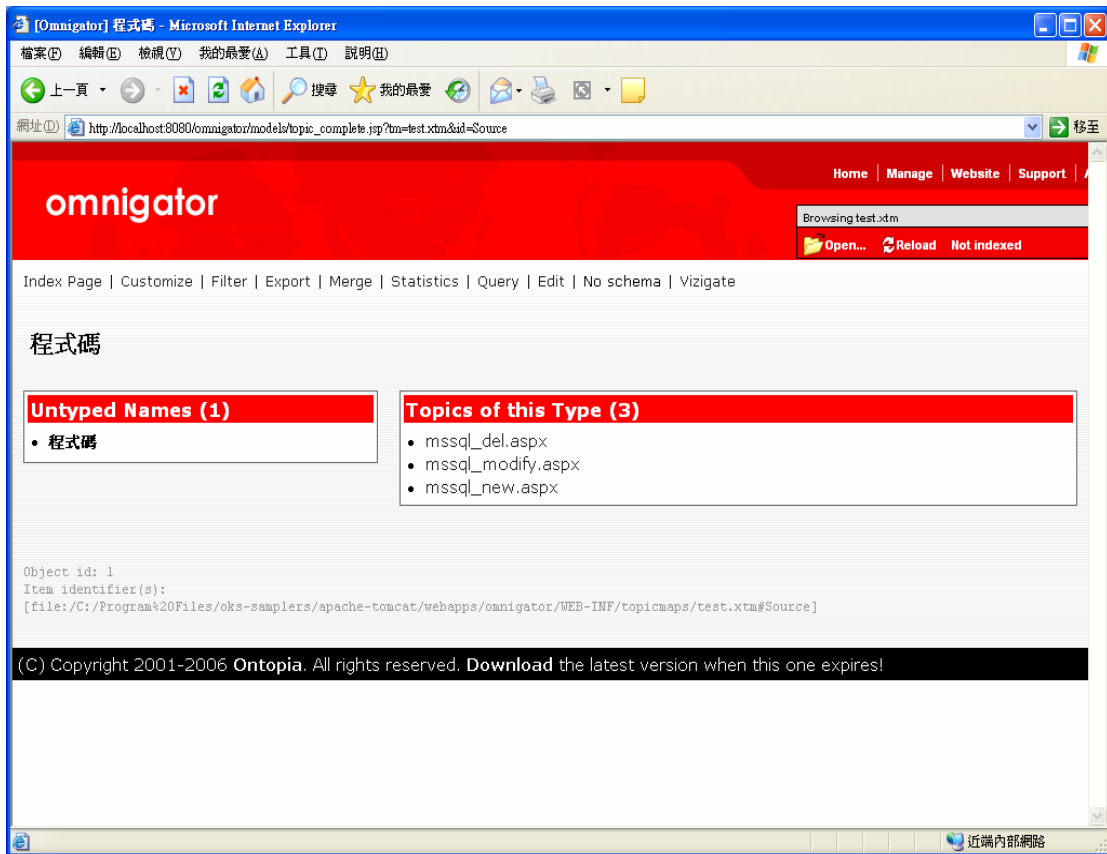


圖 5.6 進入「程式碼」主題類別

點選進入主題「mssql_modify.aspx」，如下圖5.7，畫面左半邊顯示從mssql_modify.aspx衍生的關聯：顯示mssql_modify.aspx「修改」資料表「PATENT」，且mssql_modify.aspx有一個主題類別「mssql_modify.aspx(程式碼說明)」，表示此原始碼有註釋其程式碼的說明。

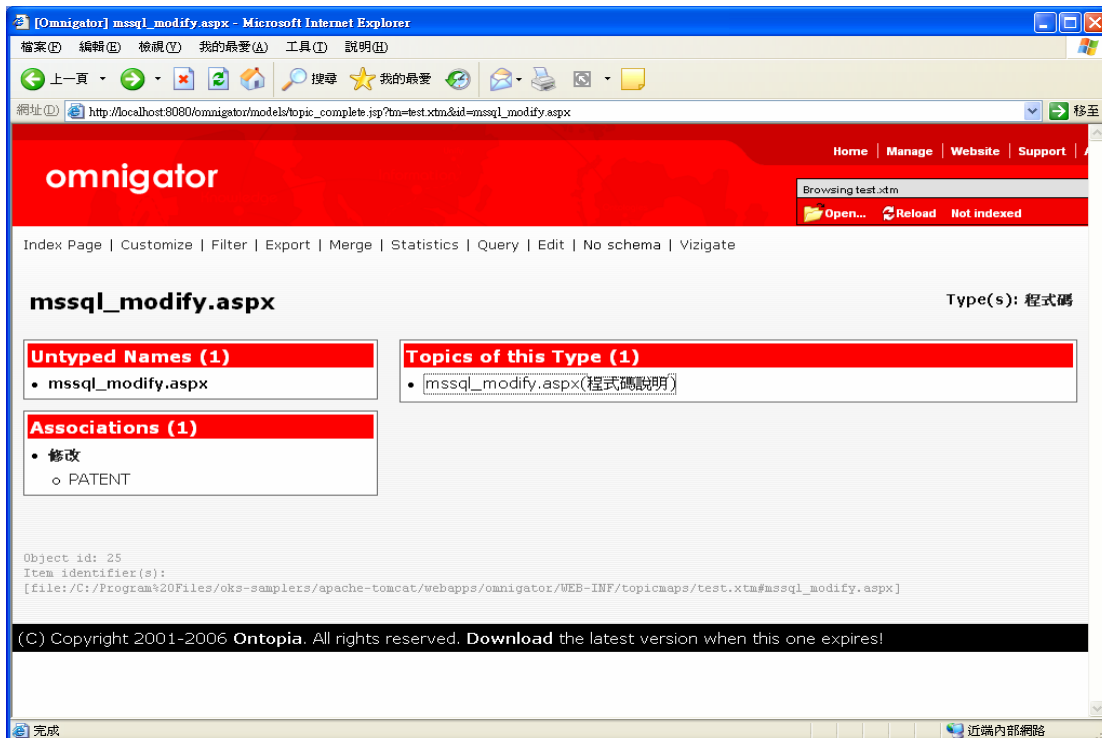


圖 5.7 進入主題「mssql_modify.aspx」

點選「mssql_modify.aspx(程式碼說明)」如下圖 5.8，顯示其程式碼說明的資源為參考外部的連結網頁。

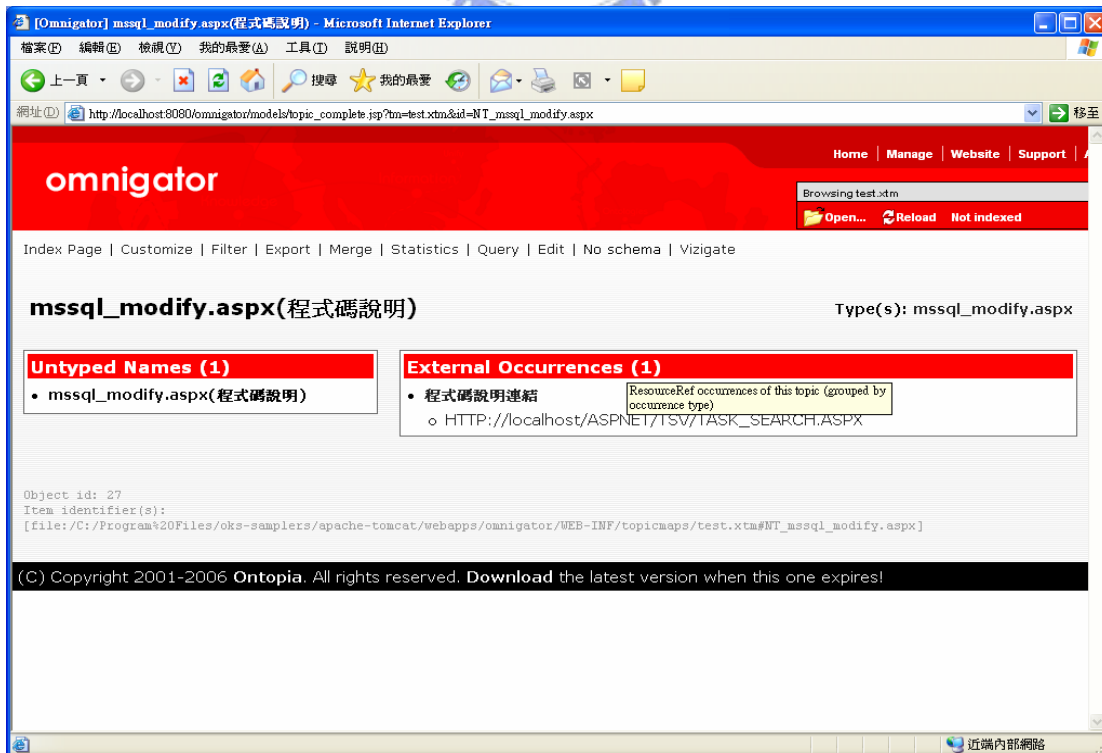


圖 5.8 進入資源「mssql_modify.aspx(程式碼說明)」

點選其關聯類別「修改」中的「資料表 (PATENT)」後，進入主題 PATENT，如下圖 5.9，由此可以看出，其位於程式「mssql_modify.aspx」，其間的關聯為「修改」；位於程式「mssql_del.aspx」，其間的關聯為「刪除」；位於程式「mssql_new.aspx」，其間的關聯為「新增」。且此資料表(PATENT)，存在有四個欄位(P_ABSTRACT, P_CLAIM, P_INVENTOR, P_TITLE)。

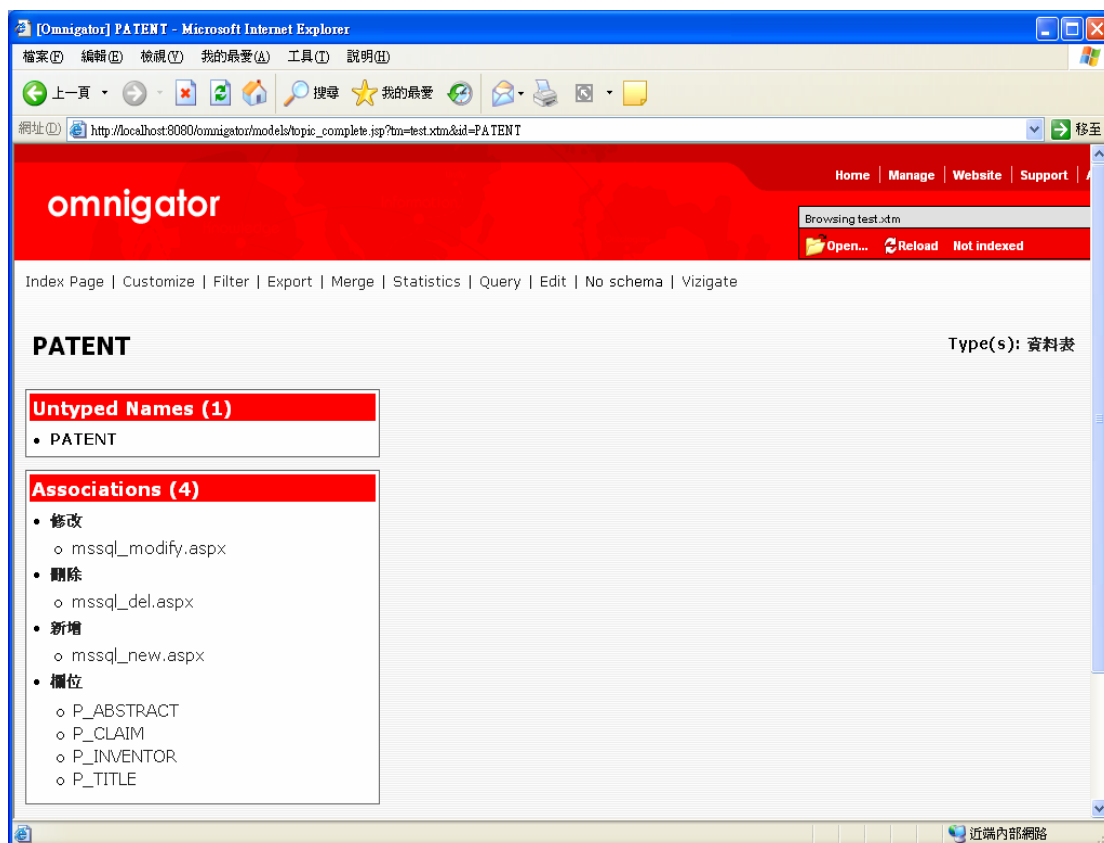


圖 5.9 進入主題「PATENT」

下圖 5.10 是點選主題「PATENT」之後開啟的視覺化主題地圖，此圖以 PATENT 為起點，可點選彈出所有和 PATENT 有關的主題，以及和這些主題相關的其他主題。

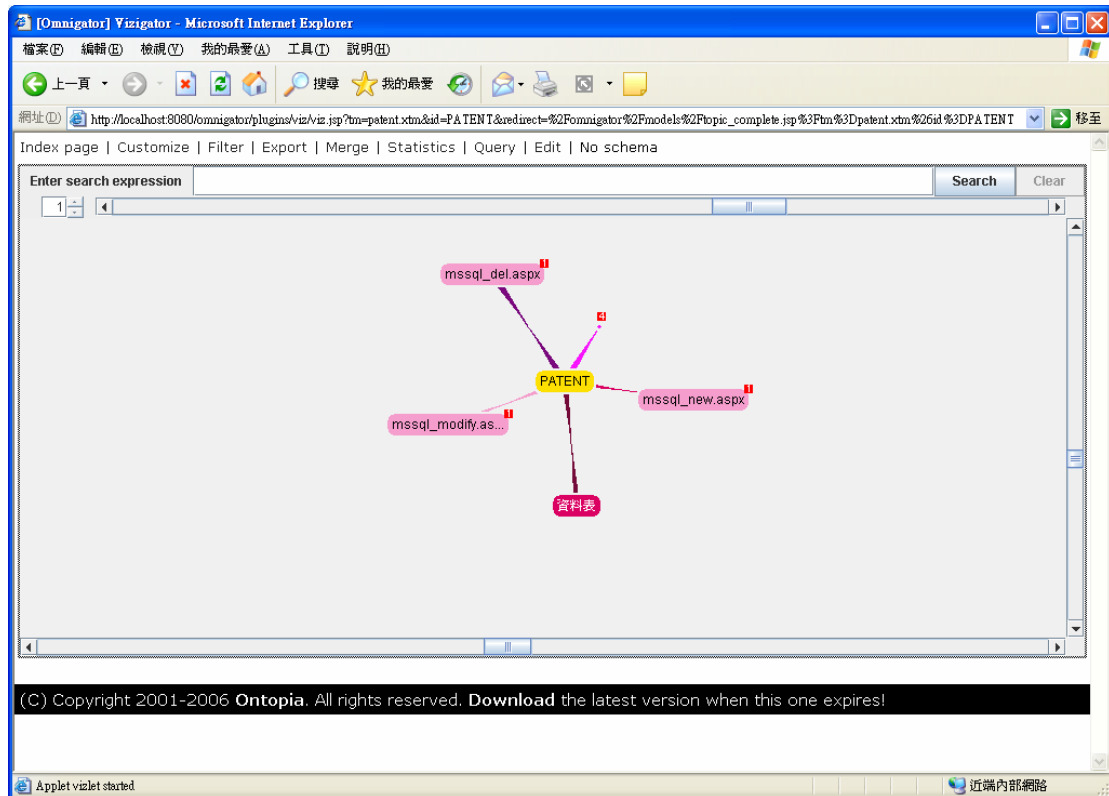


圖 5.10 以「PATENT」為起點的視覺化主題地圖

下圖 5.11 顯示與「PATENT」之關聯資訊，例如將滑鼠移至「PATENT」與「mssql_del.aspx」的連結線上時，會顯示其間的有一關聯為「刪除」，表示原始碼「mssql_del.aspx」存在「刪除」資料表「PATENT」的資訊。以相同的方式操作，可以顯示與資訊表「PATENT」其他的關聯資訊。

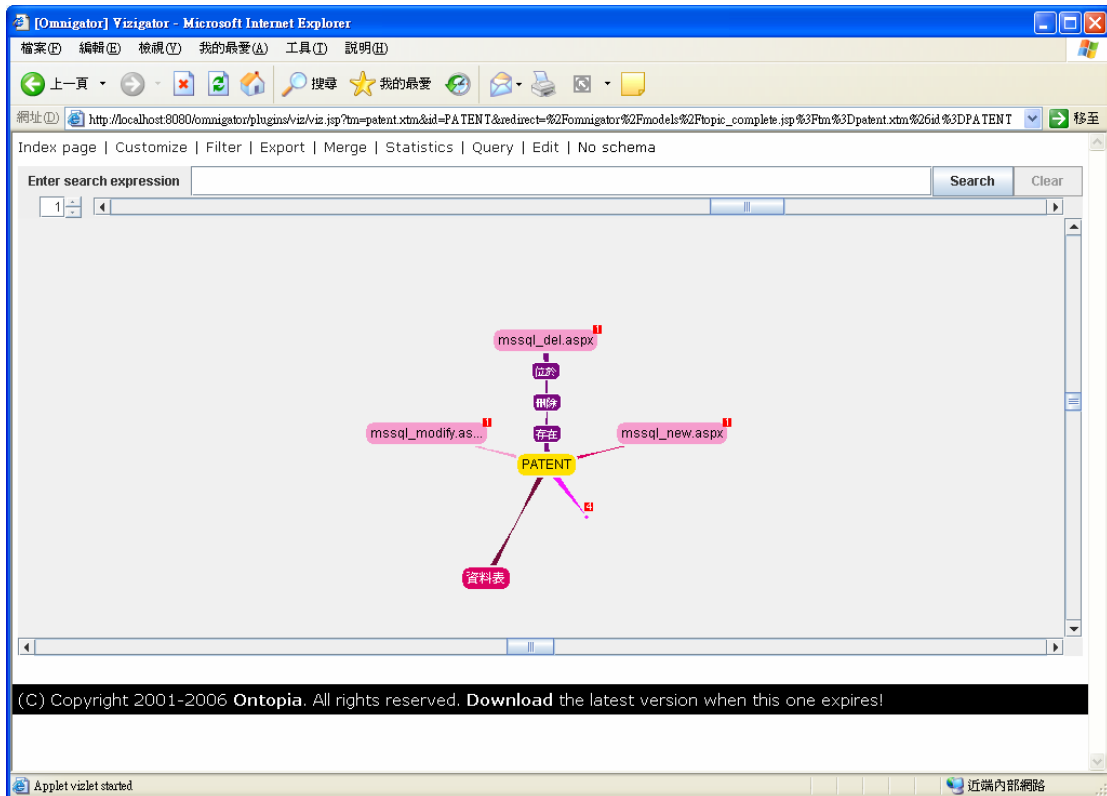


圖 5.11 顯示與「PATENT」之關聯資訊

下圖 5.12 透過 Expand Node 方式，顯示資料表「PATENT」的欄位，滑鼠移至主圖地圖中的數字 4 會顯示「欄位」，此數字與「PATENT」中間的關聯為「位於」，表示有 4 個「欄位」位於資料表「PATENT」，於此數字 4 按右鍵點選「Expand Node」，會彈出並顯示與此資料表「PATENT」的 4 個欄位，如下圖 5.13。

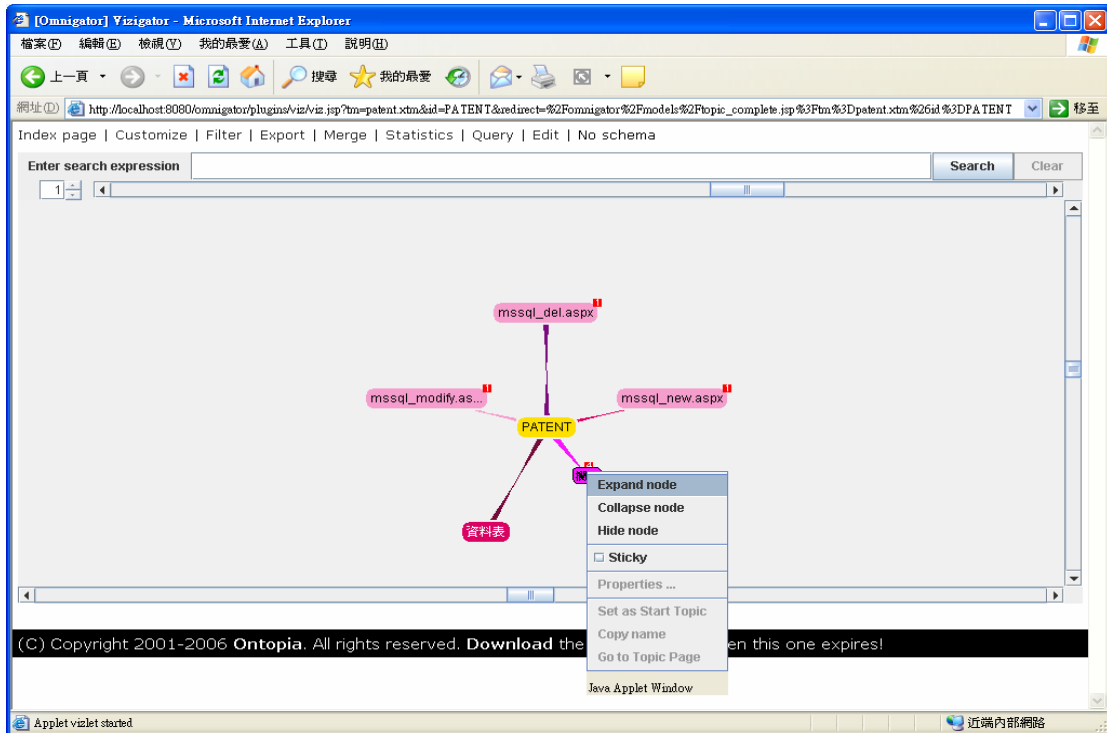


圖 5.11 透過 Expand node 顯示更多資訊

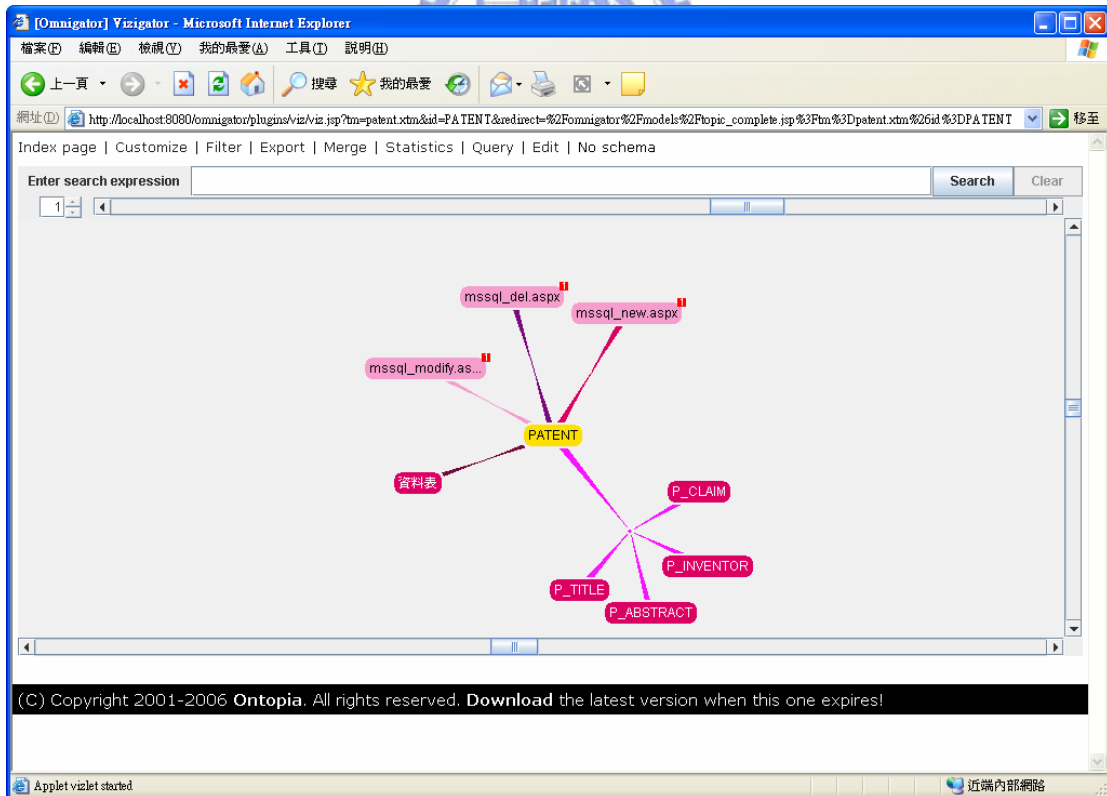


圖 5.13 顯示資料表「PATENT」的欄位

5.7 Web 查詢過濾介面

有時系統規模較大，相關的主題、關聯及資源，相對的也較為龐大，為了方便使用者快速查詢及過濾其想要的，例如，想要在上百個資料表中，找出某幾個想要新增欄位的資料表時，可使用此 Web 介面查詢並過濾其條件，快速地只列出使用者需要的相關資料。使用者介面可依系統(模組)名稱、查詢類型、存取種類及關鍵字查詢等過濾條件查詢。例如下圖 5.14，使用者依系統(模組)名稱「行事曆(calendar)」，過濾只列出資料表為「CALENDAR_NEW」的相關資料。

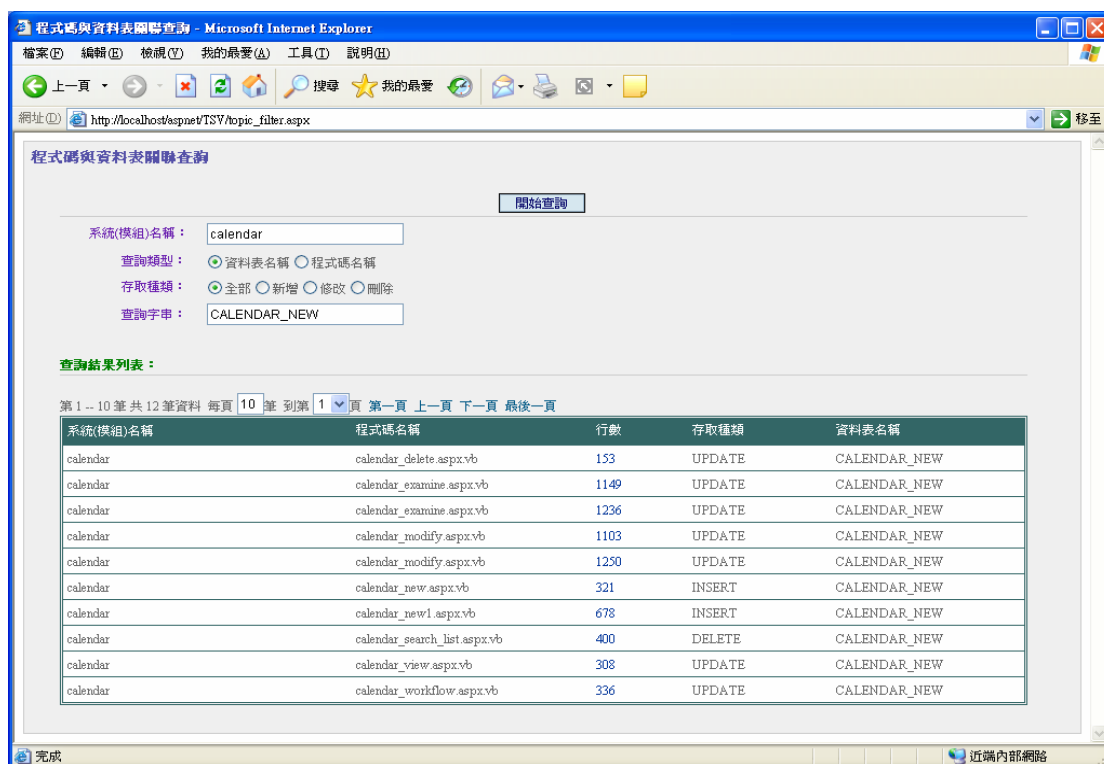


圖 5.14 程式碼與資料表關聯查詢

點選 calendar_new.aspx.vb，行數第「321」，可檢視其詳細資料，如下圖 5.15，顯示此原始碼所屬的系統(模組)名稱，程式碼來源的位置，且列出此原始碼於第 321 行新增(INSERT)資料表(CALENDAR_NEW)的欄位有 22 個。

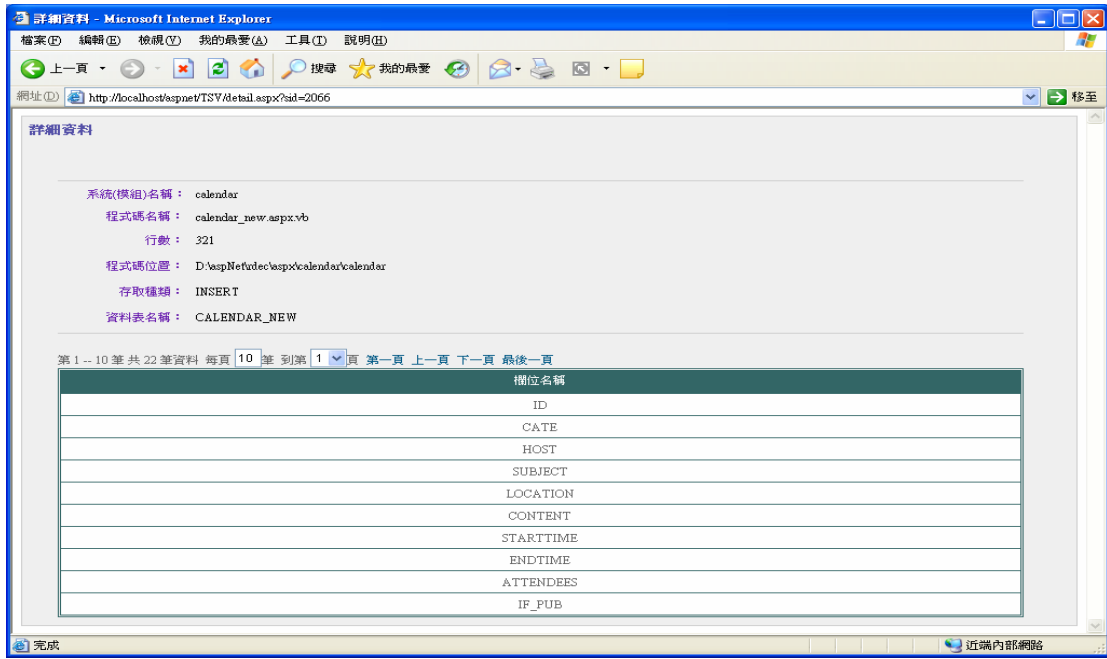


圖 5.15 詳細資料頁面



6 系統評估

系統的評估方式，採用一般系統評估常用的 Precision 與 Recall，進行系統的評估。Precision 與 Recall 示意圖如下圖 6.1。A 表示資料庫應用系統實際擁有的資料表數，R 表示本系統針下此資料庫應用系統的原始碼，剖析出來的資料表數，Ra 表示 A 與 R 的交集。Precision 與 Recall 的表示分別如下：

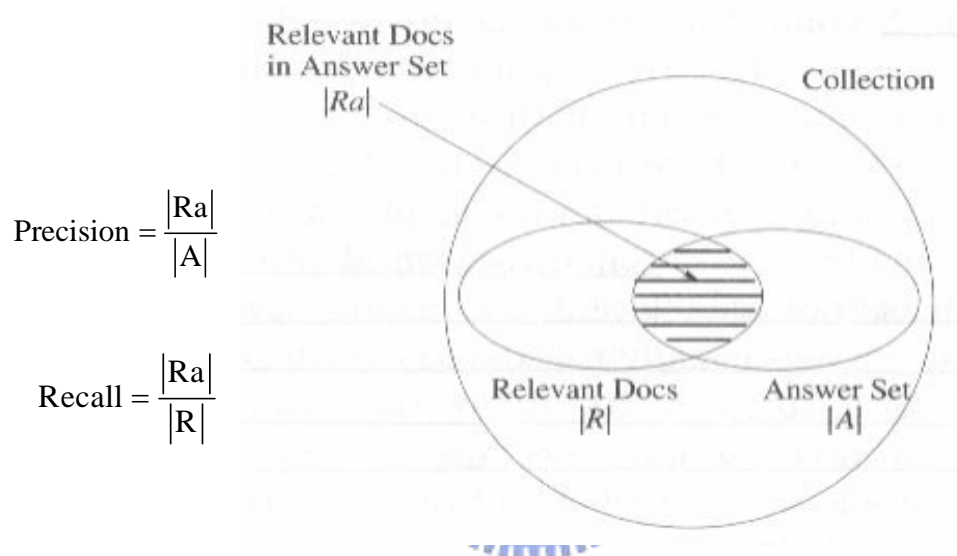


圖 6.1 Precision 與 Recall 示意圖

6.1 原始碼與資料庫操作－「新增」之評估

本系統依表 5.2 選取的資料庫應用程式，對資料庫操作中的新增，所做的評估如下表 6.1，相關資料表數為系統剖析出的資料表數，實際資料表數為此資料庫應用程式系統實際的資料表數，相關且實際的資料表數為以上兩者的交集，依此數據分別計算其 Precision 及 Recall，平均的 Precision 為 95.96%，Recall 為 100%。討論區及留言板的 Precision 未達 100% 的原因，在於其 SQL 敘述為多行時其字串連結的語法，與系統設定資料表(CONJ)的不符造成。

資料庫應用程式名稱	原始碼檔案數	相關資料表數	實際資料表數	相關且實際資料表數	Precision	Recall
儲存媒體管理系統	69	36	36	36	100%	100%
生產紀錄管理資訊系統	147	33	33	33	100%	100%
問卷調查	27	6	6	6	100%	100%
線上報名	42	5	5	5	100%	100%
討論區	39	2	3	2	66.67%	100%
意見箱	15	2	2	2	100%	100%
留言板	25	4	5	4	80%	100%
行事曆	25	2	2	2	100%	100%
公佈欄	26	7	7	7	100%	100%
工作管理	31	9	9	9	100%	100%

表 6.1：原始碼與資料庫操作(新增)之評估

6.2 原始碼與資料庫操作－「修改」之評估

本系統依表 5.2 選取的資料庫應用程式，對資料庫操作中的修改，所做的評估如下表 6.2，相關資料表數為系統剖析出的資料表數，實際資料表數為此資料庫應用程式系統實際的資料表數，相關且實際的資料表數為以上兩者的交集，依此數據分別計算其 Precision 及 Recall，平均的 Precision 為 100%，Recall 為 100%。

資料庫應用程式名稱	原始碼檔案數	相關資料表數	實際資料表數	相關且實際資料表數	Precision	Recall
儲存媒體管理系統	69	24	25	25	100%	100%
生產紀錄管理資訊系統	147	25	25	25	100%	100%
問卷調查	27	5	5	5	100%	100%
線上報名	42	5	5	5	100%	100%
討論區	39	5	5	5	100%	100%
意見箱	15	2	2	2	100%	100%
留言板	25	3	3	3	100%	100%
行事曆	25	1	1	1	100%	100%
公佈欄	26	4	4	4	100%	100%
工作管理	31	7	7	7	100%	100%

表 6.2：原始碼與資料庫操作(修改)之評估

6.3 原始碼與資料庫操作－「刪除」之評估

本系統依表 5.2 選取的資料庫應用程式，對資料庫操作中的刪除，所做的評估如下表 6.3，相關資料表數為系統剖析出的資料表數，實際資料表數為此資料庫應用程式系統實際的資料表數，相關且實際的資料表數為以上兩者的交集，依此數據分別計算其 Precision 及 Recall，平均的 Precision 為 96.97%，Recall 為 100%。問卷調查的 Precision 未達 100% 的原因，在於其 SQL 敘述內含多個刪除的敘述造成。

資料庫應用程式名稱	原始碼檔案數	相關資料表數	實際資料表數	相關且實際資料表數	Precision	Recall
儲存媒體管理系統	69	40	40	40	100%	100%
生產紀錄管理資訊系統	147	7	7	7	100%	100%
問卷調查	27	3	6	3	50%	100%
線上報名	42	5	5	5	100%	100%
討論區	39	2	2	2	100%	100%
意見箱	15	1	1	1	100%	100%
留言板	25	4	4	4	100%	100%
行事曆	25	1	1	1	100%	100%
公佈欄	26	5	5	5	100%	100%
工作管理	31	13	13	13	100%	100%

表 6.3：原始碼與資料庫操作(刪除)之評估

7 結論與未來研究方向

7.1 結論

本研究已從資料庫應用程式的最基礎來源，即程式原始碼，擷取相關存取資料的資訊，並應用主題地圖的概念，完成視覺化主題地圖的展現，將資料庫應用程式與資料庫綱要之間的關係以視覺化地圖的方式呈現。以下提出其對於軟體開發人員及測試人員所帶來的助益。

對軟體開發人員來說，當系統需要進行改版時，視覺化的主題地圖，可以讓開發人員快速且完整地瞭解程式與資料庫綱要之間的關係，可有效縮短開發的時程；當系統因應功能擴增，也可以快速掌握相關需要變更的程式進行異動，全面並正確地完成新增的需求；當系統因應需求變更，也可以有效地找出相關需要修改的程式，確保變更的程式，不會影響其他相關程式的正常運作，俾使異動應用程式後仍與資料庫能夠完全契合。因應需求快速變化的時代，讓軟體的開發與維護，達成高效率、高品質、低成本的軟體發展目標。

對測試人員來說，當程式異動時，可以快速地掌握相關的程式進行測試，以確保其影響範圍，從單元測試、整合測試至系統測試，讓各階段皆能進行完整且全面的測試，以提高軟體可靠度及軟體的品質。

7.2 未來研究方向

1、多行 SQL 敘述的連結方式

可再擴充系統設定資料表(CONJ)，使系統可辨識多種的字串連結符號，其它程式語言如 Java、PHP 等亦可以此法進行擴充剖析。

2、執行的 SQL 敘述中內含多個 SQL 敘述

可加強剖析執行的 SQL 敘述中內含多個 SQL 敘述的情形。

3、資料表及欄位使用變數

對於 SQL 敘述中使用變數的情形，目前並無法辨識，未來研究或許可搭配一些工具(如 SQL Server 的 Profiler)於執行時期(run time)，擷取執行時的 SQL 敘述進行剖析。

4、資料讀取的敘述

本研究只針對造成資料異動的資料庫操作語言，即新增(INSERT)、修改(UPDATE)及刪除(DELETE)，未來的研究可思考加入剖析資料讀取(SELECT)的 SQL 敘述。



參考文獻

- [1] Beier, J. & Tesche, T. (2000). Navigation and Interaction in Medical Knowledge Spaces Using Topic Maps. International Congress Series, 1230, 384-388.
- [2] Biezunski, M. (2003). Introduction to the Topic Maps Paradigm. In J. Park & S. Hunting (Ed.), XML Topic Maps: Creating and Using Topic Maps for the Web (pp.17-30). Boston: Addison-Wesley.
- [3] DeRose, S., Maler, E., Orchard, D. (2001). XML Linking Language (XLink) Version 1.0. Available at: <http://www.w3.org/TR/xlink/> (June 14, 2005)
- [4] Garshol, L. M., Barta, R., JTC1/SC34. (2005). Topic Maps Query Language. Available at: <http://www.isotopicmaps.org/tmql/spec.html> (May 18, 2005)
- [5] Lars Marius Garshol. What Are Topic Maps. Available at: <http://www.xml.com/pub/a/2002/09/11/topicmaps.html?page=2>.
- [6] Member of the TopicMaps.Org Authoring Group. (2001). XML Topic Maps (XTM) v1.0. Available at: <http://www.topicmaps.org/xtm/1.0/> (May19, 2005)
- [7] Moore, G., Bogachev, D., Nishikawa, M., JTC1/SC34. (2005). Topic Maps Constraint Language. Available at: <http://www.isotopicmaps.org/tmcl/tmcl-2005-02-12.html> (May 18, 2005)
- [8] Navathe & Elmasri (2003). Fundamentals of Database Systems, Fourth Edition (p.268). Addison Wesley.
- [9] Omnigator, Topic Maps Tool. Available at: <http://www.ontopia.net/omnigator/models/index.jsp>.
- [10] Pepper, S. (2000). The TAO of Topic Maps: Finding the Way in the Age of

Infoglut. Available at: <http://www.ontopia.net/topicmaps/materials/tao.html> (June 2, 2005)

[11] Pepper S., & Garshol, L. M. (2002). The XML Papers: Lessons on Applying Topic Maps. IDEAlliance XML 2002 Conference, Baltimore.

[12] TopicMaps.Org. Topic Maps standard [ISO13250]. Available at: <http://www.topicmaps.org/>.

[13] SQL 語句語法。 Available at: <http://twpug.net/docs/mysql-5.1/sql-syntax.html>

[14] SQL 簡介， Available at: http://dyna.hcc.edu.tw/php/class4_3.htm

[15] SQL 關鍵字. Available at:

<http://www.php5.idv.tw/documents/pgsqli-7.3tw/sql-keywords-appendix.html>

[16] 內嵌式 SQL. Available at:

<http://www.dbmaker.com.tw/prodinfo/dbmaker/d-teach/paper12.html>

[17] 林信成、歐陽慧、歐陽崇榮 (2004)。以主題地圖建構索引典之語意網路模型。圖書與資訊學刊，48，35-56。

[18] 卓素珍(2003)。資料存取技術發展，「網頁資料庫」教學設計與應用之研究。

[19] 施威銘研究室(2006)，SQL Server 2005 設計實務，旗標股份有限公司

[20] 胡大雄(2002)，資料庫系統開放式界面-ODBC， Available at:

<http://www.dbmaker.com.tw/reference/issue/dbmaiss5.html#pic1>

[21] 胡百敬(2006)，SQL Server 2005 資料庫開發聖經，學貫行銷股份有限公司

附錄 A：程式碼剖析及 XTM 匯出之原始碼

```
Imports System.IO
Imports System.Data.SqlClient
Imports System.Text
Public Class Form1
    Dim cn As SqlConnection
    Dim str1, str2, str4, KItem, ic, sc As String
    Dim arr1(), arr2(), arr_f1(), arr_f2(), arr_f3(), arr_tmp(), conj() As String
    Dim nLine, nLineConj, nID As Integer
    Dim Field As String
    Dim str_update, str_update2 As String
    Dim sCONJ, sConjString As String
    Dim sb As StringBuilder
    Dim sRefData, sRefUrl As String
    Private Function checkExistData(ByVal sFileName As String, ByVal sKeyWord As String,
ByVal sKItem As String, ByVal nLine As Integer, ByVal sDir As String) As Integer
        sc = ""
        If rb_type.Checked = True Then
            sc = "select * from PSCT where psc_sc=" & sFileName & " and psc_type=" &
sKeyWord & " and psc_table=" & sKItem & " and psc_line=" & nLine & ""
        Else
            sc = "select * from PSCT where psc_sc=" & sFileName & " and psc_type=" &
sKeyWord & " and psc_table=" & sKItem & " and psc_line=" & nLine & " and psc_dir=" & sDir &
""
        End If
        Dim da As SqlDataAdapter = New SqlDataAdapter(sc, cn)
        Dim ds As New DataSet
        da.Fill(ds, "PSCT")
        If ds.Tables("PSCT").Rows.Count > 0 Then
            checkExistData = ds.Tables("PSCT").Rows(0).Item("psc_id")
        Else
            checkExistData = 0
        End If
    End Function
    Private Function checkExistField(ByVal nID As Integer, ByVal sFName As String) As Boolean
        sc = ""
        sc = "select * from PSCF where f_fsn=" & nID & " and f_fname=" & sFName & ""
        Dim da As SqlDataAdapter = New SqlDataAdapter(sc, cn)
        Dim ds As New DataSet
        da.Fill(ds, "PSCT")
        If ds.Tables("PSCT").Rows.Count > 0 Then
            checkExistField = True
        Else
            checkExistField = False
        End If
    End Function
    Private Sub insertData(ByVal sFileName As String, ByVal sKeyWord As String, ByVal sKItem
As String, ByVal nLine As Integer, ByVal sDir As String, ByVal sRefData As String, ByVal sRefUrl
As String)
        ic = ""
        If rb_type.Checked = True Then
            ic = "insert into PSCT
(psc_sc,psc_type,psc_table,psc_line,psc_flag,psc_dir,psc_sysname,psc_filetype,psc_refdata,psc_refurl)
values (" & sFileName & ", " & sKeyWord & ", " & sKItem & ", " & nLine & ", 'FILE', " & sDir &
", " & tb_name.Text & ", " & cb_FileType.Text & ", " & sRefData & ", " & sRefUrl & ""
        Else
        End Sub
End Class
```

```

        ic = "insert into PSCT
(psc_sc,psc_type,psc_table,psc_line,psc_flag,psc_dir,psc_sysname,psc_filetype,psc_refdata,psc_refurl)
values (" & sFileName & "," & sKeyword & "," & sKItem & "," & nLine & ", 'DIR'," & sDir & ","
& tb_name.Text & "," & cb_FileType.Text & "," & sRefData & "," & sRefUrl & ")
    End If
    Dim cmd As SqlCommand = New SqlCommand(ic, cn)
    cmd.ExecuteNonQuery()
End Sub
Private Sub insertField(ByVal nID As Integer, ByVal sFName As String)
    ic = ""
    ic = "insert into PSCF (f_fsn,f_fname) values (" & nID & "," & sFName & ")"
    Dim cmd As SqlCommand = New SqlCommand(ic, cn)
    cmd.ExecuteNonQuery()
End Sub
Private Function getIdentity() As Integer
    Dim sc As String
    sc = "select @@identity"
    Dim da As SqlDataAdapter = New SqlDataAdapter(sc, cn)
    Dim ds As New DataSet
    da.Fill(ds, "identity")
    getIdentity = ds.Tables("identity").Rows(0).Item(0)
End Function
Private Sub parseKeyword(ByVal sStr As String, ByVal sFileName As String, ByVal nLine As
Integer, ByVal sDir As String, ByVal sRefData As String, ByVal sRefUrl As String)
    str1 = ""
    str2 = ""
    KItem = ""
    arr1 = Nothing
    arr2 = Nothing
    arr_f1 = Nothing
    arr_f2 = Nothing
    arr_f3 = Nothing
    arr_tmp = Nothing
    nID = 0
    Field = ""
    str_update = ""
    str_update2 = ""
    str1 = Trim(sStr).Trim(vbTab).ToUpper
    If Mid(str1, 1, 1) <> "" Then
        If str1.IndexOf("INSERT ") <> -1 Then
            If str1.IndexOf(" VALUES") <> -1 Then
                str2 = Trim(str1.Substring(str1.IndexOf("INSERT ") + 7)).Trim(vbTab)
                arr1 = str2.Split(" ")
                If arr1(0) = "INTO" Then
                    str4 = Trim(str2.Substring(str2.IndexOf("INTO ") + 5)).Trim(vbTab)
                    arr2 = str4.Split(" ")
                    KItem = arr2(0)
                    If KItem.IndexOf("(") <> -1 Then
                        KItem = KItem.Substring(0, KItem.IndexOf("("))
                    End If
                Else
                    KItem = arr1(0)
                End If
            If checkExistData(sFileName, "INSERT", KItem, nLine, sDir) = 0 Then
                insertData(sFileName, "INSERT", KItem, nLine, sDir, sRefData,
sRefUrl)

                nID = getIdentity()
                If nID > 0 Then

```



```

arr_f1 = str2.Split("")
If arr_f1(0).Contains(" VALUES") = False Then
    arr_f2 = Split(arr_f1(0), "(")
    arr_f3 = arr_f2(1).Split(",")
    Dim i As Integer
    For i = 0 To arr_f3.Length - 1
        Field = Trim(arr_f3(i)).Trim(vbTab)
        If checkExistField(nID, Field) = False Then
            insertField(nID, Field)
        End If
    Next
End If
End If
End If
End If
End If
arr1 = Nothing
arr2 = Nothing
arr_f1 = Nothing
arr_f2 = Nothing
arr_f3 = Nothing
arr_tmp = Nothing
If str1.IndexOf("UPDATE ") <> -1 Then
    If str1.IndexOf("SET ") <> -1 Then
        str2 = Trim(str1.Substring(str1.IndexOf("UPDATE ") + 7)).Trim(vbTab)
        If str2.Contains("SET ") Then
            arr1 = str2.Split(" ")
            KItem = arr1(0)
            If KItem.IndexOf("(") <> -1 Then
                KItem = KItem.Substring(0, KItem.IndexOf("("))
            End If
            If checkExistData(sFileName, "UPDATE", KItem, nLine, sDir) = 0 Then
                insertData(sFileName, "UPDATE", KItem, nLine, sDir, sRefData,
sRefUrl)

                nID = getIdentity()
                If nID > 0 Then
                    str_update = Trim(str2.Substring(str2.IndexOf("SET ") +
3)).Trim(vbTab)

                    If str_update.Contains(" WHERE ") Then
                        str_update2 = str_update.Substring(0,
str_update.LastIndexOf("WHERE"))
                    Else
                        str_update2 = str_update
                    End If
                    arr_f3 = str_update2.Split(",")
                    Dim i As Integer
                    For i = 0 To arr_f3.Length - 1
                        If arr_f3(i).Contains("=") Then
                            arr_tmp = arr_f3(i).Split("=")
                            Field = Trim(arr_tmp(0)).Trim(vbTab)
                            If checkExistField(nID, Field) = False Then
                                insertField(nID, Field)
                            End If
                        End If
                    Next
                End If
            End If
        End If
    End If
End If
End If

```

```

End If
arr1 = Nothing
arr2 = Nothing
arr_f1 = Nothing
arr_f2 = Nothing
arr_f3 = Nothing
arr_tmp = Nothing
Dim nPos As Integer
nPos = str1.IndexOf("DELETE ")
If nPos <> -1 Then
    If nPos - 1 > 0 Then
        If (Asc(Mid(str1, nPos - 1, 1)) < 65 Or Asc(Mid(str1, nPos - 1, 1)) > 90) And
(Asc(Mid(str1, nPos - 1, 1)) < 97 Or Asc(Mid(str1, nPos - 1, 1)) > 122) Then
            str2 = Trim(str1.Substring(str1.IndexOf("DELETE ") + 7)).Trim(vbTab)
            arr1 = str2.Split(" ")
            If arr1(0) = "FROM" Then
                str4 = Trim(str2.Substring(str2.IndexOf("FROM ") +
5)).Trim(vbTab)

                arr2 = str4.Split(" ")
                KItem = arr2(0)
            Else
                KItem = arr1(0)
            End If
            If KItem.IndexOf("(") <> -1 Then
                KItem = KItem.Substring(0, KItem.IndexOf("("))
            End If
            If KItem.IndexOf(")") <> -1 Then
                KItem = KItem.Substring(0, KItem.IndexOf(")"))
            End If
            If checkExistData(sFileName, "DELETE", KItem, nLine, sDir) = 0 Then
                insertData(sFileName, "DELETE", KItem, nLine, sDir, sRefData,
sRefUrl)
            End If
        End If
    End If
End If
End If
End If
End Sub
Private Sub parseFile()
    Dim sStr As String
    If cb_FileType.Text <> "" Then
        If cb_FileType.Text = "*" Then
            dlgOpen.Filter = cb_FileType.Text & "|" & cb_FileType.Text
        Else
            dlgOpen.Filter = cb_FileType.Text & "|*." & cb_FileType.Text
        End If
    End If
    If dlgOpen.ShowDialog() = Windows.Forms.DialogResult.OK Then
        cn.Open()
        Dim myFile As FileInfo
        myFile = New FileInfo(dlgOpen.FileName())
        Try
            Dim fs As FileStream
            fs = myFile.Open(FileMode.Open)
            Dim bCheck As Boolean = False
            Dim sr As New StreamReader(fs, System.Text.Encoding.Default)
            If fs.Length > 0 Then
                nLine = 0
            End If
        Catch
        End Try
    End If
End Sub

```

```

sRefData = ""
sRefUrl = ""
Do
    sStr = sr.ReadLine()
    nLine += 1
    If nLine = 1 Then
        Dim sTMP As String
        sTMP = Trim(sStr).Trim(vbTab).ToUpper
        Dim arrTMP() As String
        If sTMP.Contains("REFDATA~") Then
            arrTMP = sTMP.Split("~")
            sRefData = arrTMP(1)
        Else
            If sTMP.Contains("REFURL~") Then
                arrTMP = sTMP.Split("~")
                sRefUrl = arrTMP(1)
            End If
        End If
    End If
    parseKeyword(sStr, myFile.Name, nLine, myFile.DirectoryName,
sRefData, sRefUrl)
Loop While sr.Peek <> -1
End If
sr.Close()
fs.Close()
MsgBox("程式碼檔案剖析完成!", MsgBoxStyle.Information)
tb_name.Text = ""
Catch ex As Exception
    MsgBox(ex.Message, MsgBoxStyle.Exclamation)
Finally
    cn.Close()
End Try
End If
End Sub
Private Sub parsing(ByVal fFileInfo As FileInfo, ByVal sDir As String)
    Dim sStr As String
    Dim fs As FileStream
    fs = fFileInfo.Open(FileMode.Open)
    Dim bCheck As Boolean = False
    Dim sr As New StreamReader(fs, System.Text.Encoding.Default)
    If fs.Length > 0 Then
        nLine = 0
        sRefData = ""
        sRefUrl = ""
        Do
            sStr = sr.ReadLine()
            nLine += 1
            If nLine = 1 Then
                Dim sTMP As String
                sTMP = Trim(sStr).Trim(vbTab).ToUpper
                Dim arrTMP() As String
                If sTMP.Contains("REFDATA~") Then
                    arrTMP = sTMP.Split("~")
                    sRefData = arrTMP(1)
                Else
                    If sTMP.Contains("REFURL~") Then
                        arrTMP = sTMP.Split("~")
                        sRefUrl = arrTMP(1)
                    End If
                End If
            End If
        Loop While sr.Peek <> -1
    End If
    sr.Close()
    fs.Close()
    MsgBox("程式碼檔案剖析完成!", MsgBoxStyle.Information)
    tb_name.Text = ""
Catch ex As Exception
    MsgBox(ex.Message, MsgBoxStyle.Exclamation)
Finally
    cn.Close()
End Try
End If
End Sub
Private Sub parsing(ByVal fFileInfo As FileInfo, ByVal sDir As String)
    Dim sStr As String
    Dim fs As FileStream
    fs = fFileInfo.Open(FileMode.Open)
    Dim bCheck As Boolean = False
    Dim sr As New StreamReader(fs, System.Text.Encoding.Default)
    If fs.Length > 0 Then
        nLine = 0
        sRefData = ""
        sRefUrl = ""
        Do
            sStr = sr.ReadLine()
            nLine += 1
            If nLine = 1 Then
                Dim sTMP As String
                sTMP = Trim(sStr).Trim(vbTab).ToUpper
                Dim arrTMP() As String
                If sTMP.Contains("REFDATA~") Then
                    arrTMP = sTMP.Split("~")
                    sRefData = arrTMP(1)
                Else
                    If sTMP.Contains("REFURL~") Then
                        arrTMP = sTMP.Split("~")
                        sRefUrl = arrTMP(1)
                    End If
                End If
            End If
        Loop While sr.Peek <> -1
    End If
    sr.Close()
    fs.Close()
    MsgBox("程式碼檔案剖析完成!", MsgBoxStyle.Information)
    tb_name.Text = ""
Catch ex As Exception
    MsgBox(ex.Message, MsgBoxStyle.Exclamation)
Finally
    cn.Close()
End Try
End If
End Sub

```




```

        End If
    End If
    End If
    parseKeyword(sStr, fileInfo.Name, nLine, sDir, sRefData, sRefUrl)
    Loop While sr.Peek <> -1
End If
sr.Close()
fs.Close()
End Sub
Private Sub parseDir()
    If FBD.ShowDialog = Windows.Forms.DialogResult.OK Then
        cn.Open()
        Try
            Dim mySrcDir As DirectoryInfo
            mySrcDir = New DirectoryInfo(FBD.SelectedPath)
            Dim myFile() As FileInfo
            myFile = mySrcDir.GetFiles()
            Dim n As Integer
            For n = 0 To myFile.Length - 1
                If cb_FileType.Text = " *.* " Then
                    parsing(myFile(n), mySrcDir.FullName)
                Else
                    If myFile(n).Extension = "." & cb_FileType.Text Then
                        parsing(myFile(n), mySrcDir.FullName)
                    End If
                End If
            Next
            MsgBox("程式碼目錄下之檔案剖析完成!", MsgBoxStyle.Information)
        Catch ex As Exception
            MsgBox(ex.Message)
        Finally
            cn.Close()
        End Try
    End If
End Sub
Private Function checkInput() As Boolean
    If tb_name.Text = "" Then
        MsgBox("請輸入系統(模組)名稱!", MsgBoxStyle.Information)
        tb_name.Focus()
        Return False
    End If
    If cb_FileType.SelectedItem Is Nothing And cb_FileType.Text = "" Then
        MsgBox("請選取程式碼檔案種類!", MsgBoxStyle.Information)
        cb_FileType.Focus()
        Return False
    End If
    Return True
End Function
Private Function checkInput2() As Boolean
    If tb_xtm.Text = "" Then
        MsgBox("請輸入系統(模組)名稱!", MsgBoxStyle.Information)
        tb_xtm.Focus()
        Return False
    End If
    Return True
End Function
Private Function getSConj(ByVal c_sType As String) As String
    Dim sc As String

```



```

sc = "select * from CONJ where c_sType='" & c_sType & "'"
Dim da As SqlDataAdapter = New SqlDataAdapter(sc, cn)
Dim ds As New DataSet
da.Fill(ds, "CONJ")
If ds.Tables("CONJ").Rows.Count > 0 Then
    If IsDBNull(ds.Tables("CONJ").Rows(0).Item("c_symbol")) Then
        getSConj = ""
    Else
        getSConj = ds.Tables("CONJ").Rows(0).Item("c_symbol")
    End If
Else
    getSConj = ""
End If
End Function
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
    If checkInput() Then
        If rb_type.Checked = True Then
            parseFile()
        Else
            parseDir()
        End If
    End If
End Sub
Private Sub getFileType()
Dim sc As String
sc = "select * from conj order by c_sType"
Dim da As SqlDataAdapter = New SqlDataAdapter(sc, cn)
Dim ds As New DataSet
da.Fill(ds, "conj")
Dim i As Integer
cb_FileType.Items.Clear()
For i = 0 To ds.Tables("conj").Rows.Count - 1
    cb_FileType.Items.Add(ds.Tables("conj").Rows(i).Item("c_sType"))
Next
End Sub
Private Sub Form1_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Me.Load
    cn = New
SqlConnection("Server=.\SQLEXPRESS;Database=PSC;Trusted_Connection=True;")
cn.Open()
Try
    getFileType()
Catch ex As Exception
    MsgBox(ex.Message, MsgBoxStyle.Exclamation)
Finally
    cn.Close()
End Try
End Sub
Private Sub preXTM()
sb = New StringBuilder
sb.Append("<?xml version=""1.0"" encoding=""utf-8"" standalone=""yes""?>")
sb.Append("<topicMap xmlns=""http://www.topicmaps.org/xtm/1.0/""
xmlns:xlink=""http://www.w3.org/1999/xlink"" id=""SCPE"">")
sb.Append("<topic id=""Source"">")
sb.Append("<baseName>")
sb.Append("<baseNameString>程式碼</baseNameString>")
sb.Append("</baseName>")

```

```

sb.Append("</topic>")
sb.Append("<topic id=""Table"">")
sb.Append("<baseName>")
sb.Append("<baseNameString>資料表</baseNameString>")
sb.Append("</baseName>")
sb.Append("</topic>")
sb.Append("<topic id=""INSERT"">")
sb.Append("<baseName>")
sb.Append("<baseNameString>新增</baseNameString>")
sb.Append("</baseName>")
sb.Append("</topic>")
sb.Append("<topic id=""UPDATE"">")
sb.Append("<baseName>")
sb.Append("<baseNameString>修改</baseNameString>")
sb.Append("</baseName>")
sb.Append("</topic>")
sb.Append("<topic id=""DELETE"">")
sb.Append("<baseName>")
sb.Append("<baseNameString>刪除</baseNameString>")
sb.Append("</baseName>")
sb.Append("</topic>")
sb.Append("<topic id=""locate"">")
sb.Append("<baseName>")
sb.Append("<baseNameString>位於</baseNameString>")
sb.Append("</baseName>")
sb.Append("</topic>")
sb.Append("<topic id=""exist"">")
sb.Append("<baseName>")
sb.Append("<baseNameString>存在</baseNameString>")
sb.Append("</baseName>")
sb.Append("</topic>")
sb.Append("<topic id=""Field"">")
sb.Append("<baseName>")
sb.Append("<baseNameString>欄位</baseNameString>")
sb.Append("</baseName>")
sb.Append("</topic>")
sb.Append("<topic id=""Note"">")
sb.Append("<baseName>")
sb.Append("<baseNameString>程式碼說明</baseNameString>")
sb.Append("</baseName>")
sb.Append("</topic>")
sb.Append("<topic id=""NoteURL"">")
sb.Append("<baseName>")
sb.Append("<baseNameString>程式碼說明連結</baseNameString>")
sb.Append("</baseName>")
sb.Append("</topic>")
sb.Append("<topic id=""SourcePath"">")
sb.Append("<baseName>")
sb.Append("<baseNameString>程式碼來源路徑</baseNameString>")
sb.Append("</baseName>")
sb.Append("</topic>")

```

End Sub

```

Private Sub writeXTM(ByVal SysName As String, ByVal sbXTM As StringBuilder)
    Dim sw As New StreamWriter("C:\" & SysName & ".xtm")
    sw.Write(sbXTM.ToString)
    sw.Close()

```

End Sub

```

Private Sub exportXTM(ByVal SysName As String)

```

```

preXTM()
Dim sc As String
sc = "select distinct psc_sc from psct where psc_sysname=" & SysName & ""
Dim da As SqlDataAdapter = New SqlDataAdapter(sc, cn)
Dim ds As New DataSet
da.Fill(ds, "psc_sc")
Dim i, j, k As Integer
For i = 0 To ds.Tables("psc_sc").Rows.Count - 1
    sb.Append("<topic id="" & ds.Tables("psc_sc").Rows(i).Item("psc_sc") & "">")
    sb.Append("<instanceOf>")
    sb.Append("<topicRef xlink:href=""#Source""/>")
    sb.Append("</instanceOf>")
    sb.Append("<baseName>")
    sb.Append("<baseNameString> & ds.Tables("psc_sc").Rows(i).Item("psc_sc") &
"</baseNameString>")
    sb.Append("</baseName>")
    sb.Append("</topic>")
    sc = "select psc_refdata,psc_refurl,psc_dir,psc_sc from psct where psc_sysname=" &
SysName & "" and psc_sc=" & ds.Tables("psc_sc").Rows(i).Item("psc_sc") & ""
    Dim da_ref As SqlDataAdapter = New SqlDataAdapter(sc, cn)
    Dim ds_ref As New DataSet
    da_ref.Fill(ds_ref, "ref")
    If ds_ref.Tables("ref").Rows.Count > 0 Then
        If ds_ref.Tables("ref").Rows(0).Item("psc_refdata") <> "" Then
            sb.Append("<topic id=""NT_ & ds_ref.Tables("ref").Rows(0).Item("psc_sc")
& "">")
            sb.Append("<instanceOf>")
            sb.Append("<topicRef xlink:href=""#"" &
ds_ref.Tables("ref").Rows(0).Item("psc_sc") & ""/>")
            sb.Append("</instanceOf>")
            sb.Append("<baseName>")
            sb.Append("<baseNameString> &
ds_ref.Tables("ref").Rows(0).Item("psc_sc") & "(程式碼說明)" & "</baseNameString>")
            sb.Append("</baseName>")
            sb.Append("<occurrence>")
            sb.Append("<instanceOf>")
            sb.Append("<topicRef xlink:href=""#Note""/>")
            sb.Append("</instanceOf>")
            sb.Append("<resourceData>")
            sb.Append(ds_ref.Tables("ref").Rows(0).Item("psc_refdata"))
            sb.Append("</resourceData>")
            sb.Append("</occurrence>")
            sb.Append("</topic>")
        End If
        If ds_ref.Tables("ref").Rows(0).Item("psc_refurl") <> "" Then
            sb.Append("<topic id=""NT_ & ds_ref.Tables("ref").Rows(0).Item("psc_sc")
& "">")
            sb.Append("<instanceOf>")
            sb.Append("<topicRef xlink:href=""#"" &
ds_ref.Tables("ref").Rows(0).Item("psc_sc") & ""/>")
            sb.Append("</instanceOf>")
            sb.Append("<baseName>")
            sb.Append("<baseNameString> &
ds_ref.Tables("ref").Rows(0).Item("psc_sc") & "(程式碼說明)" & "</baseNameString>")
            sb.Append("</baseName>")
            sb.Append("<occurrence>")
            sb.Append("<instanceOf>")
            sb.Append("<topicRef xlink:href=""#NoteURL""/>")

```

```

        sb.Append("</instanceOf>")
        sb.Append("<resourceRef xlink:href=""")
        sb.Append(" & ds_ref.Tables("ref").Rows(0).Item("psc_refUrl") & """)
        sb.Append(""/>")
        sb.Append("</occurrence>")
        sb.Append("</topic>")

    End If
End If
Next
sc = "select distinct psc_table from psct where psc_sysname="" & SysName & ""
Dim da2 As SqlDataAdapter = New SqlDataAdapter(sc, cn)
da2.Fill(ds, "psc_table")
For i = 0 To ds.Tables("psc_table").Rows.Count - 1
    sb.Append("<topic id="" & ds.Tables("psc_table").Rows(i).Item("psc_table") & "">")
    sb.Append("<instanceOf>")
    sb.Append("<topicRef xlink:href=""#Table""/>")
    sb.Append("</instanceOf>")
    sb.Append("<baseName>")
    sb.Append("<baseNameString> & ds.Tables("psc_table").Rows(i).Item("psc_table") &
"</baseNameString>")
    sb.Append("</baseName>")
    sb.Append("</topic>")
    sc = "select distinct f_fname from pscf where f_fsn in (select psc_id from psct where
psc_sysname="" & SysName & "" and psc_table="" &
ds.Tables("psc_table").Rows(i).Item("psc_table") & "" order by f_fname"
    Dim da3 As SqlDataAdapter = New SqlDataAdapter(sc, cn)
    Dim ds3 As New DataSet
    da3.Fill(ds3, "fld")
    If ds3.Tables("fld").Rows.Count > 0 Then
        For j = 0 To ds3.Tables("fld").Rows.Count - 1
            sb.Append("<topic id="" & ds3.Tables("fld").Rows(j).Item("f_fname") &
"">")
            sb.Append("<baseName>")
            sb.Append("<baseNameString> & ds3.Tables("fld").Rows(j).Item("f_fname")
& "</baseNameString>")
            sb.Append("</baseName>")
            sb.Append("</topic>")
        Next
        sb.Append("<association>")
        sb.Append("<instanceOf>")
        sb.Append("<topicRef xlink:href=""#Field""/>")
        sb.Append("</instanceOf>")
        sb.Append("<member>")
        sb.Append("<roleSpec>")
        sb.Append("<topicRef xlink:href=""#locate""/>")
        sb.Append("</roleSpec>")
        sb.Append("<topicRef xlink:href=""#" &
ds.Tables("psc_table").Rows(i).Item("psc_table") & "">")
        sb.Append("</member>")
        sb.Append("<member>")
        sb.Append("<roleSpec>")
        sb.Append("<topicRef xlink:href=""#exist""/>")
        sb.Append("</roleSpec>")
        For j = 0 To ds3.Tables("fld").Rows.Count - 1
            sb.Append("<topicRef xlink:href=""#" &
ds3.Tables("fld").Rows(j).Item("f_fname") & "">")
        Next

```

```

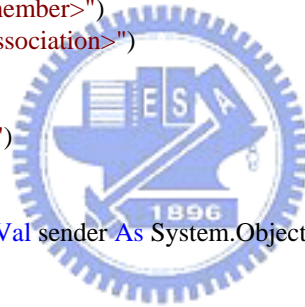
        sb.Append("</member>")
        sb.Append("</association>")
    End If
    sc = "select distinct psc_sc from psct where psc_type='INSERT' and psc_sysname=" &
SysName & "' and psc_table=" & ds.Tables("psc_table").Rows(i).Item("psc_table") & "'"
    Dim da4 As SqlDataAdapter = New SqlDataAdapter(sc, cn)
    Dim ds4 As New DataSet
    da4.Fill(ds4, "asso")
    If ds4.Tables("asso").Rows.Count > 0 Then
        sb.Append("<association>")
        sb.Append("<instanceOf>")
        sb.Append("<topicRef xlink:href=""#INSERT""/>")
        sb.Append("</instanceOf>")
        sb.Append("<member>")
        sb.Append("<roleSpec>")
        sb.Append("<topicRef xlink:href=""#locate""/>")
        sb.Append("</roleSpec>")
        For k = 0 To ds4.Tables("asso").Rows.Count - 1
            sb.Append("<topicRef xlink:href=""#" &
ds4.Tables("asso").Rows(k).Item("psc_sc") & """/>")
            Next
            sb.Append("</member>")
            sb.Append("<member>")
            sb.Append("<roleSpec>")
            sb.Append("<topicRef xlink:href=""#exist""/>")
            sb.Append("</roleSpec>")
            sb.Append("<topicRef xlink:href=""#" &
ds.Tables("psc_table").Rows(i).Item("psc_table") & """/>")
            sb.Append("</member>")
            sb.Append("</association>")
        End If
        sc = "select distinct psc_sc from psct where psc_type='UPDATE' and psc_sysname=" &
SysName & "' and psc_table=" & ds.Tables("psc_table").Rows(i).Item("psc_table") & "'"
        Dim da5 As SqlDataAdapter = New SqlDataAdapter(sc, cn)
        da5.Fill(ds4, "asso_u")
        If ds4.Tables("asso_u").Rows.Count > 0 Then
            sb.Append("<association>")
            sb.Append("<instanceOf>")
            sb.Append("<topicRef xlink:href=""#UPDATE""/>")
            sb.Append("</instanceOf>")
            sb.Append("<member>")
            sb.Append("<roleSpec>")
            sb.Append("<topicRef xlink:href=""#locate""/>")
            sb.Append("</roleSpec>")
            For k = 0 To ds4.Tables("asso_u").Rows.Count - 1
                sb.Append("<topicRef xlink:href=""#" &
ds4.Tables("asso_u").Rows(k).Item("psc_sc") & """/>")
                Next
                sb.Append("</member>")
                sb.Append("<member>")
                sb.Append("<roleSpec>")
                sb.Append("<topicRef xlink:href=""#exist""/>")
                sb.Append("</roleSpec>")
                sb.Append("<topicRef xlink:href=""#" &
ds.Tables("psc_table").Rows(i).Item("psc_table") & """/>")
                sb.Append("</member>")
                sb.Append("</association>")
            End If
    End If

```

```

        sc = "select distinct psc_sc from psct where psc_type='DELETE' and psc_sysname=" &
SysName & "' and psc_table=" & ds.Tables("psc_table").Rows(i).Item("psc_table") & "'"
        Dim da6 As SqlDataAdapter = New SqlDataAdapter(sc, cn)
        da6.Fill(ds4, "asso_d")
        If ds4.Tables("asso_d").Rows.Count > 0 Then
            sb.Append("<association>")
            sb.Append("<instanceOf>")
            sb.Append("<topicRef xlink:href=""#DELETE""/>")
            sb.Append("</instanceOf>")
            sb.Append("<member>")
            sb.Append("<roleSpec>")
            sb.Append("<topicRef xlink:href=""#locate""/>")
            sb.Append("</roleSpec>")
            For k = 0 To ds4.Tables("asso_d").Rows.Count - 1
                sb.Append("<topicRef xlink:href=""#" &
ds4.Tables("asso_d").Rows(k).Item("psc_sc") & """/>")
                Next
            sb.Append("</member>")
            sb.Append("<member>")
            sb.Append("<roleSpec>")
            sb.Append("<topicRef xlink:href=""#exist""/>")
            sb.Append("</roleSpec>")
            sb.Append("<topicRef xlink:href=""#" &
ds.Tables("psc_table").Rows(i).Item("psc_table") & """/>")
            sb.Append("</member>")
            sb.Append("</association>")
        End If
    Next
    sb.Append("</topicMap>")
    writeXTM(SysName, sb)
End Sub
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button2.Click
    If checkInput2() Then
        cn.Open()
        Try
            exportXTM(tb_xtm.Text)
            MsgBox("匯出XTM檔案完成!", MsgBoxStyle.Information)
        Catch ex As Exception
            MsgBox(ex.Message, MsgBoxStyle.Exclamation)
        Finally
            cn.Close()
        End Try
    End If
End Sub
End Class

```



附錄 B：Web 查詢過濾介面之原始碼

B.1 查詢列表頁

```
Imports System.Data
Imports System.Data.SqlClient
Public Class task_search
    Inherits System.Web.UI.Page
    Dim cn As New SqlConnection(ConfigurationSettings.AppSettings("ConnString"))
    Protected WithEvents ButtonSearch As System.Web.UI.WebControls.Button
    Protected WithEvents HyperLinkSearchDetail As System.Web.UI.WebControls.HyperLink
    Protected WithEvents DataGrid1 As System.Web.UI.WebControls.DataGrid
    Protected WithEvents LabelResult As System.Web.UI.WebControls.Label
    Protected WithEvents lblMessage As System.Web.UI.WebControls.Label
    Protected WithEvents RecordStatus As System.Web.UI.WebControls.Label
    Protected WithEvents cnt As System.Web.UI.WebControls.TextBox
    Protected WithEvents dd1 As System.Web.UI.WebControls.DropDownList
    Protected WithEvents lb1 As System.Web.UI.WebControls.LinkButton
    Protected WithEvents lb2 As System.Web.UI.WebControls.LinkButton
    Protected WithEvents lb3 As System.Web.UI.WebControls.LinkButton
    Protected WithEvents lb4 As System.Web.UI.WebControls.LinkButton
    Protected WithEvents Panel1 As System.Web.UI.WebControls.Panel
    Protected WithEvents lb_nodata As System.Web.UI.WebControls.Label
    Protected WithEvents tb_keyword As System.Web.UI.WebControls.TextBox
    Protected WithEvents tb_sysname As System.Web.UI.WebControls.TextBox
    Protected WithEvents rbl_type As System.Web.UI.WebControls.RadioButtonList
    Protected WithEvents rbl_access As System.Web.UI.WebControls.RadioButtonList
    Dim selectCmd As String
    Dim tableTask As DataTable
    Sub ChangePerRecord(ByVal sender As Object, ByVal e As EventArgs)
        If IsNumeric(cnt.Text) Then
            DataGrid1.CurrentPageIndex = 0
            DataGrid1.EditItemIndex = -1
            Call bindSearch()
        Else
            lblMessage.Text = "請輸入數字!"
            lblMessage.ForeColor = Color.Red
            lblMessage.Font.Bold = True
        End If
    End Sub
    Sub PageChange(ByVal sender As Object, ByVal e As CommandEventArgs)
        If e.CommandArgument.ToString = "first" Then '第一頁
            DataGrid1.CurrentPageIndex = 0
        ElseIf e.CommandArgument.ToString = "last" Then '最後一頁
            DataGrid1.CurrentPageIndex = DataGrid1.PageCount - 1
        ElseIf e.CommandArgument.ToString = "pre" Then '上一頁
            If DataGrid1.CurrentPageIndex > 0 Then
                DataGrid1.CurrentPageIndex -= 1
            End If
        ElseIf e.CommandArgument.ToString = "next" Then '下一頁
            If DataGrid1.CurrentPageIndex < DataGrid1.PageCount - 1 Then
                DataGrid1.CurrentPageIndex += 1
            End If
        End If
        DataGrid1.EditItemIndex = -1
        Call bindSearch()
    End Sub
End Class
```



```

End Sub
Private Sub ddl1_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles ddl1.SelectedIndexChanged
    DataGrid1.CurrentPageIndex = Integer.Parse(ddl1.SelectedItem.Value) - 1
    DataGrid1.EditItemIndex = -1
    Call bindSearch()
End Sub
Public Sub getSelectCmd()
    If tb_sysname.Text <> "" Then
        selectCmd &= " and psc_sysname like '%" & tb_sysname.Text & "%'"
    End If
    If tb_keyword.Text <> "" Then
        If rbl_type.SelectedValue = "table" Then
            selectCmd &= " and psc_table like '%" & tb_keyword.Text & "%'"
        Else
            selectCmd &= " and psc_sc like '%" & tb_keyword.Text & "%'"
        End If
    End If
    If rbl_access.SelectedValue <> "ALL" Then
        selectCmd &= " and psc_type = '" & rbl_access.SelectedValue & "'"
    End If
End Sub
Public Function bindSearch() As DataTable
    selectCmd = "select * from psct where psc_id>0 "
    Call getSelectCmd()
    Dim da As SqlDataAdapter = New SqlDataAdapter(selectCmd, cn)
    Dim ds As New DataSet
    da.Fill(ds, "psct")
    tableTask = ds.Tables("psct")
    If tableTask.Rows.Count > 0 Then
        LabelResult.Visible = True
        DataGrid1.Visible = True
        Panell.Visible = True
        lb_nodata.Visible = False
        DataGrid1.DataSource = tableTask
        DataGrid1.DataBind()
        DataGrid1.PageSize = Integer.Parse(cnt.Text)
        Dim page As New page_fun
        page.PageFun(ddl1, DataGrid1, cnt, tableTask, RecordStatus)
    Else
        LabelResult.Visible = True
        DataGrid1.Visible = False
        Panell.Visible = False
        lb_nodata.Visible = True
        lb_nodata.Text = "查無符合您需求的資料!!"
    End If
End Function
Private Sub ButtonSearch_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles ButtonSearch.Click
    cn.Open()
    Try
        Call bindSearch()
    Catch ex As Exception
        lblMessage.Text = ex.Message
    Finally
        cn.Close()
    End Try
End Sub

```



```

Private Sub DataGrid1_DataBinding(ByVal sender As Object, ByVal e As System.EventArgs)
Handles DataGrid1.DataBinding
    If DataGrid1.CurrentPageIndex > tableTask.Rows.Count \ cnt.Text Then
        DataGrid1.CurrentPageIndex = 0
    End If
End Sub
End Class

```

B.2 詳細內容頁

```

Imports System.Data
Imports System.Data.SqlClient
Public Class detail
    Inherits System.Web.UI.Page
    Protected WithEvents lblMessage As System.Web.UI.WebControls.Label
    Protected WithEvents RecordStatus As System.Web.UI.WebControls.Label
    Protected WithEvents cnt As System.Web.UI.WebControls.TextBox
    Protected WithEvents dd1 As System.Web.UI.WebControls.DropDownList
    Protected WithEvents lb1 As System.Web.UI.WebControls.LinkButton
    Protected WithEvents lb2 As System.Web.UI.WebControls.LinkButton
    Protected WithEvents lb3 As System.Web.UI.WebControls.LinkButton
    Protected WithEvents lb4 As System.Web.UI.WebControls.LinkButton
    Protected WithEvents Panel1 As System.Web.UI.WebControls.Panel
    Protected WithEvents DataGrid1 As System.Web.UI.WebControls.DataGrid
    Protected WithEvents l_sysname As System.Web.UI.WebControls.Label
    Protected WithEvents l_sc As System.Web.UI.WebControls.Label
    Protected WithEvents l_table As System.Web.UI.WebControls.Label
    Protected WithEvents l_line As System.Web.UI.WebControls.Label
    Protected WithEvents l_access As System.Web.UI.WebControls.Label
    Protected WithEvents l_path As System.Web.UI.WebControls.Label
    Private designerPlaceholderDeclaration As System.Object
    Dim cn As New SqlConnection(ConfigurationSettings.AppSettings("ConnString"))
    Sub ChangePerRecord(ByVal sender As Object, ByVal e As EventArgs)
        If IsNumeric(cnt.Text) Then
            DataGrid1.CurrentPageIndex = 0
            DataGrid1.EditItemIndex = -1
            Call bindSearch(Request("sid"))
        Else
            lblMessage.Text = "請輸入數字!"
            lblMessage.ForeColor = Color.Red
            lblMessage.Font.Bold = True
        End If
    End Sub
    Sub PageChange(ByVal sender As Object, ByVal e As CommandEventArgs)
        If e.CommandArgument.ToString = "first" Then '第一頁
            DataGrid1.CurrentPageIndex = 0
        ElseIf e.CommandArgument.ToString = "last" Then '最後一頁
            DataGrid1.CurrentPageIndex = DataGrid1.PageCount - 1
        ElseIf e.CommandArgument.ToString = "pre" Then '上一頁
            If DataGrid1.CurrentPageIndex > 0 Then
                DataGrid1.CurrentPageIndex -= 1
            End If
        ElseIf e.CommandArgument.ToString = "next" Then '下一頁
            If DataGrid1.CurrentPageIndex < DataGrid1.PageCount - 1 Then
                DataGrid1.CurrentPageIndex += 1
            End If
        End Sub

```

```

        End If
        DataGrid1.EditItemIndex = -1
        Call bindSearch(Request("sid"))
    End Sub
    Private Sub ddl1_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles ddl1.SelectedIndexChanged
        DataGrid1.CurrentPageIndex = Integer.Parse(ddl1.SelectedItem.Value) - 1
        DataGrid1.EditItemIndex = -1
        Call bindSearch(Request("sid"))
    End Sub
    Private Sub bindSearch(ByVal sid As Integer)
        Dim sc As String
        sc = "select * from PSCT where psc_id=" & sid & ""
        Dim da2 As SqlDataAdapter = New SqlDataAdapter(sc, cn)
        Dim ds As New DataSet
        da2.Fill(ds, "main")
        If ds.Tables("main").Rows.Count > 0 Then
            l_sysname.Text = ds.Tables("main").Rows(0).Item("psc_sysname")
            l_sc.Text = ds.Tables("main").Rows(0).Item("psc_sc")
            l_table.Text = ds.Tables("main").Rows(0).Item("psc_table")
            l_line.Text = ds.Tables("main").Rows(0).Item("psc_line")
            l_access.Text = ds.Tables("main").Rows(0).Item("psc_type")
            l_path.Text = ds.Tables("main").Rows(0).Item("psc_dir")
            sc = "select * from PSCF where f_fsn=" & sid & ""
            Dim da As SqlDataAdapter = New SqlDataAdapter(sc, cn)
            da.Fill(ds, "psct")
            Dim tableTask As DataTable = ds.Tables("psct")
            If tableTask.Rows.Count > 0 Then
                DataGrid1.Visible = True
                Panell.Visible = True
                DataGrid1.DataSource = tableTask
                DataGrid1.PageSize = Integer.Parse(cnt.Text)
                DataGrid1.DataBind()
                Dim page As New page_fun
                page.PageFun(ddl1, DataGrid1, cnt, tableTask, RecordStatus)
            Else
                DataGrid1.Visible = False
                Panell.Visible = False
            End If
        End If
    End Sub
    Private Sub Page_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
 MyBase.Load
        If Not Page.IsPostBack Then
            bindSearch(Request("sid"))
        End If
    End Sub
End Class

```

附錄 C：XTM 檔案範例

此範例為程式碼剖析引擎模組的匯出功能所產生的檔案，摘錄資料庫應用程式-行事曆的範例。

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<topicMap xmlns="http://www.topicmaps.org/xtm/1.0/"
xmlns:xlink="http://www.w3.org/1999/xlink" id="SCPE">
  <topic id="Source">
    <baseName>
      <baseNameString>程式碼</baseNameString>
    </baseName>
  </topic>
  <topic id="Table">
    <baseName>
      <baseNameString>資料表</baseNameString>
    </baseName>
  </topic>
  <topic id="INSERT">
    <baseName>
      <baseNameString>新增</baseNameString>
    </baseName>
  </topic>
  <topic id="UPDATE">
    <baseName>
      <baseNameString>修改</baseNameString>
    </baseName>
  </topic>
  <topic id="DELETE">
    <baseName>
      <baseNameString>刪除</baseNameString>
    </baseName>
  </topic>
  <topic id="locate">
    <baseName>
      <baseNameString>位於</baseNameString>
    </baseName>
  </topic>
  <topic id="exist">
    <baseName>
      <baseNameString>存在</baseNameString>
    </baseName>
  </topic>
  <topic id="Field">
    <baseName>
      <baseNameString>欄位</baseNameString>
    </baseName>
  </topic>
  <topic id="Note">
    <baseName>
      <baseNameString>程式碼說明</baseNameString>
    </baseName>
  </topic>
  <topic id="NoteURL">
    <baseName>
      <baseNameString>程式碼說明連結</baseNameString>
    </baseName>
  </topic>

```



```

</topic>
<topic id="SourcePath">
  <baseName>
    <baseNameString>程式碼來源路徑</baseNameString>
  </baseName>
</topic>
<topic id="calendar_delete.aspx.vb">
  <instanceOf>
    <topicRef xlink:href="#Source" />
  </instanceOf>
  <baseName>
    <baseNameString>calendar_delete.aspx.vb</baseNameString>
  </baseName>
</topic>
<topic id="calendar_examine.aspx.vb">
  <instanceOf>
    <topicRef xlink:href="#Source" />
  </instanceOf>
  <baseName>
    <baseNameString>calendar_examine.aspx.vb</baseNameString>
  </baseName>
</topic>
<topic id="calendar_modify.aspx.vb">
  <instanceOf>
    <topicRef xlink:href="#Source" />
  </instanceOf>
  <baseName>
    <baseNameString>calendar_modify.aspx.vb</baseNameString>
  </baseName>
</topic>
<topic id="calendar_new.aspx.vb">
  <instanceOf>
    <topicRef xlink:href="#Source" />
  </instanceOf>
  <baseName>
    <baseNameString>calendar_new.aspx.vb</baseNameString>
  </baseName>
</topic>
<topic id="calendar_new1.aspx.vb">
  <instanceOf>
    <topicRef xlink:href="#Source" />
  </instanceOf>
  <baseName>
    <baseNameString>calendar_new1.aspx.vb</baseNameString>
  </baseName>
</topic>
<topic id="calendar_search_list.aspx.vb">
  <instanceOf>
    <topicRef xlink:href="#Source" />
  </instanceOf>
  <baseName>
    <baseNameString>calendar_search_list.aspx.vb</baseNameString>
  </baseName>
</topic>
<topic id="calendar_view.aspx.vb">
  <instanceOf>
    <topicRef xlink:href="#Source" />
  </instanceOf>

```



```

    <baseName>
      <baseNameString>calendar_view.aspx.vb</baseNameString>
    </baseName>
  </topic>
<topic id="calendar_workflow.aspx.vb">
  <instanceOf>
    <topicRef xlink:href="#Source" />
  </instanceOf>
  <baseName>
    <baseNameString>calendar_workflow.aspx.vb</baseNameString>
  </baseName>
</topic>
<topic id="calendar_workflow_modify.aspx.vb">
  <instanceOf>
    <topicRef xlink:href="#Source" />
  </instanceOf>
  <baseName>
    <baseNameString>calendar_workflow_modify.aspx.vb</baseNameString>
  </baseName>
</topic>
<topic id="CALENDAR_LOG">
  <instanceOf>
    <topicRef xlink:href="#Table" />
  </instanceOf>
  <baseName>
    <baseNameString>CALENDAR_LOG</baseNameString>
  </baseName>
</topic>
<topic id="CHECK_STATUS">
  <baseName>
    <baseNameString>CHECK_STATUS</baseNameString>
  </baseName>
</topic>
<topic id="ID">
  <baseName>
    <baseNameString>ID</baseNameString>
  </baseName>
</topic>
<topic id="MODIFY_DATE">
  <baseName>
    <baseNameString>MODIFY_DATE</baseNameString>
  </baseName>
</topic>
<topic id="MODIFY_USER">
  <baseName>
    <baseNameString>MODIFY_USER</baseNameString>
  </baseName>
</topic>
<association>
  <instanceOf>
    <topicRef xlink:href="#Field" />
  </instanceOf>
  <member>
    <roleSpec>
      <topicRef xlink:href="#locate" />
    </roleSpec>
    <topicRef xlink:href="#CALENDAR_LOG" />
  </member>

```



```

<member>
  <roleSpec>
    <topicRef xlink:href="#exist" />
  </roleSpec>
  <topicRef xlink:href="#CHECK_STATUS" />
  <topicRef xlink:href="#ID" />
  <topicRef xlink:href="#MODIFY_DATE" />
  <topicRef xlink:href="#MODIFY_USER" />
</member>
</association>
<association>
  <instanceOf>
    <topicRef xlink:href="#INSERT" />
  </instanceOf>
  <member>
    <roleSpec>
      <topicRef xlink:href="#locate" />
    </roleSpec>
    <topicRef xlink:href="#calendar_examine.aspx.vb" />
    <topicRef xlink:href="#calendar_modify.aspx.vb" />
    <topicRef xlink:href="#calendar_new.aspx.vb" />
    <topicRef xlink:href="#calendar_new1.aspx.vb" />
    <topicRef xlink:href="#calendar_workflow_modify.aspx.vb" />
  </member>
  <member>
    <roleSpec>
      <topicRef xlink:href="#exist" />
    </roleSpec>
    <topicRef xlink:href="#CALENDAR_LOG" />
  </member>
</association>
<topic id="CALENDAR_NEW">
  <instanceOf>
    <topicRef xlink:href="#Table" />
  </instanceOf>
  <baseName>
    <baseNameString>CALENDAR_NEW</baseNameString>
  </baseName>
</topic>
<topic id="APPLY_DATE">
  <baseName>
    <baseNameString>APPLY_DATE</baseNameString>
  </baseName>
</topic>
<topic id="APPLY_USER">
  <baseName>
    <baseNameString>APPLY_USER</baseNameString>
  </baseName>
</topic>
<topic id="ATTENDEES">
  <baseName>
    <baseNameString>ATTENDEES</baseNameString>
  </baseName>
</topic>
<topic id="ATTENDEES_OUT">
  <baseName>
    <baseNameString>ATTENDEES_OUT</baseNameString>
  </baseName>

```



```

</topic>
<topic id="CATE">
  <baseName>
    <baseNameString>CATE</baseNameString>
  </baseName>
</topic>
<topic id="CHECK_STATUS">
  <baseName>
    <baseNameString>CHECK_STATUS</baseNameString>
  </baseName>
</topic>
<topic id="CHECKER">
  <baseName>
    <baseNameString>CHECKER</baseNameString>
  </baseName>
</topic>
<topic id="CONTACTS_PERSON">
  <baseName>
    <baseNameString>CONTACTS_PERSON</baseNameString>
  </baseName>
</topic>
<topic id="CONTENT">
  <baseName>
    <baseNameString>CONTENT</baseNameString>
  </baseName>
</topic>
<topic id="DEP">
  <baseName>
    <baseNameString>DEP</baseNameString>
  </baseName>
</topic>
<topic id="DISPLAYNAME">
  <baseName>
    <baseNameString>DISPLAYNAME</baseNameString>
  </baseName>
</topic>
<topic id="ENDTIME">
  <baseName>
    <baseNameString>ENDTIME</baseNameString>
  </baseName>
</topic>
<topic id="HOST">
  <baseName>
    <baseNameString>HOST</baseNameString>
  </baseName>
</topic>
<topic id="ID">
  <baseName>
    <baseNameString>ID</baseNameString>
  </baseName>
</topic>
<topic id="IF_DEL">
  <baseName>
    <baseNameString>IF_DEL</baseNameString>
  </baseName>
</topic>
<topic id="IF_PUB">
  <baseName>

```




```

    <baseNameString>IF_PUB</baseNameString>
  </baseName>
</topic>
<topic id="INSTANCE">
  <baseName>
    <baseNameString>INSTANCE</baseNameString>
  </baseName>
</topic>
<topic id="LOCATION">
  <baseName>
    <baseNameString>LOCATION</baseNameString>
  </baseName>
</topic>
<topic id="MODIFY_TIME">
  <baseName>
    <baseNameString>MODIFY_TIME</baseNameString>
  </baseName>
</topic>
<topic id="MODIFY_USER">
  <baseName>
    <baseNameString>MODIFY_USER</baseNameString>
  </baseName>
</topic>
<topic id="PUB_TO">
  <baseName>
    <baseNameString>PUB_TO</baseNameString>
  </baseName>
</topic>
<topic id="STARTTIME">
  <baseName>
    <baseNameString>STARTTIME</baseNameString>
  </baseName>
</topic>
<topic id="SUBJECT">
  <baseName>
    <baseNameString>SUBJECT</baseNameString>
  </baseName>
</topic>
<topic id="UNIT">
  <baseName>
    <baseNameString>UNIT</baseNameString>
  </baseName>
</topic>
<topic id="UUID">
  <baseName>
    <baseNameString>UUID</baseNameString>
  </baseName>
</topic>
<association>
  <instanceOf>
    <topicRef xlink:href="#Field" />
  </instanceOf>
  <member>
    <roleSpec>
      <topicRef xlink:href="#locate" />
    </roleSpec>
    <topicRef xlink:href="#CALENDAR_NEW" />
  </member>

```



```

<member>
  <roleSpec>
    <topicRef xlink:href="#exist" />
  </roleSpec>
  <topicRef xlink:href="#APPLY_DATE" />
  <topicRef xlink:href="#APPLY_USER" />
  <topicRef xlink:href="#ATTENDEES" />
  <topicRef xlink:href="#ATTENDEES_OUT" />
  <topicRef xlink:href="#CATE" />
  <topicRef xlink:href="#CHECK_STATUS" />
  <topicRef xlink:href="#CHECKER" />
  <topicRef xlink:href="#CONTACTS_PERSON" />
  <topicRef xlink:href="#CONTENT" />
  <topicRef xlink:href="#DEP" />
  <topicRef xlink:href="#DISPLAYNAME" />
  <topicRef xlink:href="#ENDTIME" />
  <topicRef xlink:href="#HOST" />
  <topicRef xlink:href="#ID" />
  <topicRef xlink:href="#IF_DEL" />
  <topicRef xlink:href="#IF_PUB" />
  <topicRef xlink:href="#INSTANCE" />
  <topicRef xlink:href="#LOCATION" />
  <topicRef xlink:href="#MODIFY_TIME" />
  <topicRef xlink:href="#MODIFY_USER" />
  <topicRef xlink:href="#PUB_TO" />
  <topicRef xlink:href="#STARTTIME" />
  <topicRef xlink:href="#SUBJECT" />
  <topicRef xlink:href="#UNIT" />
  <topicRef xlink:href="#UUID" />
</member>
</association>
<association>
  <instanceOf>
    <topicRef xlink:href="#INSERT" />
  </instanceOf>
  <member>
    <roleSpec>
      <topicRef xlink:href="#locate" />
    </roleSpec>
    <topicRef xlink:href="#calendar_new.aspx.vb" />
    <topicRef xlink:href="#calendar_new1.aspx.vb" />
  </member>
  <member>
    <roleSpec>
      <topicRef xlink:href="#exist" />
    </roleSpec>
    <topicRef xlink:href="#CALENDAR_NEW" />
  </member>
</association>
<association>
  <instanceOf>
    <topicRef xlink:href="#UPDATE" />
  </instanceOf>
  <member>
    <roleSpec>
      <topicRef xlink:href="#locate" />
    </roleSpec>
    <topicRef xlink:href="#calendar_delete.aspx.vb" />
  </member>

```



```

<topicRef xlink:href="#calendar_examine.aspx.vb"/>
<topicRef xlink:href="#calendar_modify.aspx.vb"/>
<topicRef xlink:href="#calendar_view.aspx.vb"/>
<topicRef xlink:href="#calendar_workflow.aspx.vb"/>
<topicRef xlink:href="#calendar_workflow_modify.aspx.vb"/>
</member>
<member>
  <roleSpec>
    <topicRef xlink:href="#exist"/>
  </roleSpec>
  <topicRef xlink:href="#CALENDAR_NEW"/>
</member>
</association>
<association>
  <instanceOf>
    <topicRef xlink:href="#DELETE"/>
  </instanceOf>
  <member>
    <roleSpec>
      <topicRef xlink:href="#locate"/>
    </roleSpec>
    <topicRef xlink:href="#calendar_search_list.aspx.vb"/>
  </member> <member>
    <roleSpec>
      <topicRef xlink:href="#exist"/>
    </roleSpec>
    <topicRef xlink:href="#CALENDAR_NEW"/>
  </member>
</association>
</topicMap>

```

