

行政院國家科學委員會專題研究計畫 成果報告

XML 與關聯式資料庫之間資料模式和查詢語言的轉換互動與
績效評估工作量模式設計之研究(2/2)

計畫類別：個別型計畫

計畫編號：NSC93-2416-H-004-002-

執行期間：93年08月01日至94年07月31日

執行單位：國立政治大學資訊管理學系

計畫主持人： 譚家蘭

報告類型：完整報告

處理方式：本計畫可公開查詢

中 華 民 國 94 年 7 月 21 日

行政院國家科學委員會補助專題研究計畫 成果報告
 期中進度報告

XML 與關聯式資料庫之間資料模式與查詢語言的轉換互動與績效

評估工作量模式設計之研究(2/2)

計畫類別： 個別型計畫 整合型計畫

計畫編號：NSC 93-2416-H-004-002-

執行期間：93 年 08 月 01 日至 94 年 07 月 31 日

計畫主持人：譙家蘭

共同主持人：

計畫參與人員：

成果報告類型(依經費核定清單規定繳交)： 精簡報告 完整報告

本成果報告包括以下應繳交之附件：

- 赴國外出差或研習心得報告一份
- 赴大陸地區出差或研習心得報告一份
- 出席國際學術會議心得報告及發表之論文各一份
- 國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、
列管計畫及下列情形者外，得立即公開查詢

涉及專利或其他智慧財產權， 一年 二年後可公開查詢

執行單位：國立政治大學資訊管理學系

中 華 民 國 94 年 7 月 31 日

XML 與關聯式資料庫之間資料模式與查詢語言的轉換互動

與績效評估工作量模式設計之研究(2/2)

譚家蘭 博士

國立政治大學資訊管理學系教授

(02)29387692

jljan@nccu.edu.tw

摘要

隨著全球資訊網(World Wild Web , WWW , Web)的日趨普及，我們發現愈來愈多的資料是直接從網路上存取。不同於過去關連式資料庫(Relational Database Management Systems , RDBMS)的結構式資料(Structured Data)，現今許多資料是直接以 HTML (Hypertext Markup Language)格式存取。這些屬於半結構(Semistructured)或非結構式(Unstructured)的資料，我們稱之為超文件/超媒體資料(Hypertext/Hypermedia)。不同來源的資料諸如關連式資料庫和超文件資料，儲存於各自不同的資料庫，且有其各自的資料模式與查詢語言來管理與維護這些資料。面對這些異質(Heterogeneous)的資料來源，我們面臨如何做資料整合(Data Integration)，亦即允許對多元資料來源資料的存取(Multi-Sources Data Access)、資料交換(Data Exchange)，以使不同來源的資料能夠相互運算(Interoperability)，達成異質性資料的同質化(Homogeneity)。更進一步達到多元異質資料來源的可攜性(Portability)、延展性(Scalability)、相容性(Compatibility)與通透性(Transparency)。如何解決這些問題，讓不同來源的資料可雙向的互相溝通，以達到異質資料整合、資料交換和運算，將是本研究主要研究的課題。

關鍵字：延伸式標記語言 XML、半結構化資料模式、關聯資料模式、資料整合、互動運算、異質資料資源、全球資訊網 WWW

Abstract

With the popularity of WWW (World Wide Web or Web), we have seen massive volume of data emerges on the Web. Unlike traditional data stored in RDBMS (Relational Database Managements Systems) that is structured, Web data now has to be stored in the form of HTML (Hypertext Makeup Language) pages. In addition, in order to represent data and interchange between Web sources, XML (Extensible Markup Language) is fast becoming the dominant exchange standard. Like HTML, XML is a subset of SGML. However, unlike HTML tags only serve the primary purpose of describing how to display a data item, XML tags describe the data itself. The initial impetus for XML may have been primarily to enhance the ability of remote application and to interpret and operate on documents fetched over Internet. XML has become the best solution to resolve these problems with data exchange and transformation between multiple and heterogeneous Web sources. However, XML raises a fundamental research issue, i.e. how to integrate XML documents with the traditional data stored. The objective of this research is to enable the bi-directional data exchange and operation interoperability between RDBMS and XML. Furthermore, the heterogeneous Web data can become portable and scalable. Interchange and interoperability can be reused. In this research, we develop a full transformation between RDBMS and XML. We develop a dual model of XML/RM and XML/SQL interoperability. We develop a set of four components in the method to tackle the research issue. They are (1) data definition transformation (2) data manipulation transformation (3) single document transformation (4) multiple document transformation. A prototype is implemented in this research to prove the feasibility and performance of the new method.

Keywords: Extensible Markup Language, Semi-structured Data Model, Relational Data Model, Data Integration and Exchange, Interoperability, Heterogeneous Data Sources, World Wide Web

壹、緒論

一、研究動機

1. 網際網路上資料整合的問題

隨著全球資訊網(World Wide Web, WWW, Web)的日趨普及，我們發現愈來愈多的資料是直接從網路上存取。不同於過去關連式資料庫(Relational Database Management Systems, RDBMS)的結構式資料(Structured Data)，現今許多資料是直接以 HTML (Hypertext Markup Language)格式存取。這些屬於半結構(Semistructured)或非結構式(Unstructured)的資料，我們稱之為超文件/超媒體資料(Hypertext/Hypermedia)。不同來源的資料諸如關連式資料庫和超文件資料，儲存於各自不同的資料庫，且有其各自的資料模式與查詢語言來管理與維護這些資料。面對這些異質(Heterogeneous)的資料來源，我們面臨如何做資料整合(Data Integration)，亦即允許對多元資料來源資料的存取(Multi-Sources Data Access)、資料交換(Data Exchange)，以使不同來源的資料能夠相互運算(Interoperability)，達成異質性資料的同質化(Homogeneity)。更進一步達到多元異質資料來源的可攜性(Portability)、延展性(Scalability)、相容性(Compatibility)與通透性(Transparency)。如何解決這些問題，讓不同來源的資料可雙向的互相溝通，以達到異質資料整合、資料交換和運算，將是本研究主要研究的課題。

2. XML 興起

XML(Extensible Markup Language)，延伸式的標記語言，為 SGML(Standard Generalized Markup Language)的子集合。不同於 HTML 的標籤(Tag)資料的呈現，XML 其標籤是資料的定義。定義好的資料直接透過網際網路傳達於組織間，以及世界各地。資料以 XML 表達格式比以 HTML 格式表達更具意義，因為具有擷取、分析及在組織間再使用的能力。根據 IDC 和 Gartner Group 預測，到 2004-2005 年，80%在網際網路上所傳送的訊息和文件都是 XML。XML 因此成為資料整合與轉換的主要標準規範和解決方式。

二、研究議題

然而面對現存的傳統和主流商業關連式資料庫資料來源，又該如何與新興的 XML 文件結合，以達到資料真正的整合和運算。不同的資料來源現都已有其各自發展成形且運作不錯的資料模式與查詢語言，如何讓不同的資料模式做一個對應，使得資料可由一種資料模式順利轉換成另外一種資料模式，並同步更新以達到真正異質資料來源的交換與整合。關連式資料庫如 Oracle、DB2、Sybase，已

漸漸提供其儲存 XML 文件的方法，而原生性的 XML 資料庫如 Tamino 也逐漸受到重視。但其在 XML 文件與關聯式資料溝通的部分，皆尚未達到即時和雙向轉換和更新及查詢三項合一的功能和運作。因此本研究即針對傳統關聯式資料庫與 XML 文件在資料交換時，資料模式與查詢語言相互對應的溝通和運算，以期達到異質資料相互交流並進一步相互運算和同步更新的效能。

三、研究目的

目前研究與資料庫產品，對上述的問題解決方案並不完整。在技術與理論上，其對單向的資料模式轉換(XML 轉為關聯資料模式)在綱要演進(Schema Evolution)及複雜資料操縱上(Complex Query)，常顯不足或是僅以特例和人工方式來解決轉換問題，至於在雙向資料模式轉換上，加上綱要演進及資料操縱更是少見[9] [11] [28] [7]。關聯模式轉為 XML 模式與 XML 模式轉為關聯模式一系列完整的雙向模式轉換，及搭配其綱要演進與查詢語言的轉換較少完整解決。實務上在 XML 與關聯資料表轉換間的解決方案，常常是透過物件定義和中介程式去執行單一方向的轉換，轉換後也無法由對目的資料模式的演進修改，直接反應在來源資料模式上，且對於查詢語言的同時轉換也無法達成。因此本研究研發關聯式資料來源與 XML 文件的資料來源，在資料模式與查詢語言間雙向的互相對應與轉換並同步進行相互運算和資料更新，以達到異質資訊來源的資料交換與整合。

貳、文獻探討

隨著全球資訊網的日漸普及，已使得愈來愈多的資訊資源可直接於網路上取得。不同於前述所提的關聯式資料庫為結構化的資訊，這類的資料來源，為半結構化資訊。研究提出一些查詢語言處理半結構化資訊，包括 WebSQL, W3QS, WebLog, Lorel, WebOQL, UnQL, STRUDEL, ARANEUS, FLORID 等。這些研究大部分都有發展其自己的資料模式與查詢語言來描述半結構化資訊資料結構以作資料的儲存與處理，及達成對文件資料的查詢、運算與分析。本節回顧整體性及學名性半結構化和非結構化的資訊整合研究文獻，以作為本研究方法的基礎，並且作為與本研究比較的基礎。

一、WebSQL

WebSQL 為 1996 年由多倫多大學與以色列的 Tel Aviv 大學所合作開發完成的雛型系統[26]。此系統主要解決兩個問題，第一為搜尋引擎的問題，欲解決目前使用者必須先瞭解各搜尋引擎的所在位置、優缺點、與查詢介面才能作查詢的窘

境；第二為解決目前的搜尋引擎無法對文件及超連結的結構做查詢的問題。

1. 資料模式

WebSQL的特色是以虛擬圖形(Virtual Graph)的概念來表達其資料模式，將WWW看成一圖形的資料結構，並定義正規的語意(Formal Semantics)來查詢其圖形結構。類似關連式資料庫，其假設無限集合D的資料值，與有限集合T的簡單資料型態，其中T的定義域為D的子集合。節點(Node)與連結(Link)兩種值組型態定義的屬性名不同，第一種值組型態稱為節點物件(Node Object)，第二種稱為連結物件(Link Object)。在全球資訊網的模式中，文件(Documents)會對應到節點物件，超文件間的連結則會對應到連結物件，而Oid則是其單一資源位置(Uniform Resource Locator, URL)。

2. 查詢語言

WebSQL 是一個類似 SQL 對 WWW 做查詢的語言。每個 Web 物件都是以其 URL 來辨別，並且只取其最基本的屬性如標題、本文、資料格式、長度以及最後修改日期等，讓使用者利用這些資訊來查詢資料。為了查詢的目的，他們視一個 Web 物件為一個虛擬關連(Virtual Relation)中的一個值組。

二、Lorel

Lorel 語言是由史丹佛大學開發研究而成，其雛型 Lore 資料庫管理系統的查詢語言，為一查詢半結構化資料的查詢語言[1]。Lorel 語言是以 SQL/OQL 型式展現的簡明易懂的查詢語言，它可說是 ODMG 的 OQL 的延伸。Lorel 語言主要兩個創新的地方在於：(1)強制型態的使用，延伸 OQL 對半結構化資料型態使用的限制，此能力是讓不同的資料型態可做轉換以方便運算比較；(2)強大的路徑表達(Path Expression)能力，允許以彈性的型式來宣告導覽存取(Navigational Access)，此能力特別適合當使用者不知資料的真實結構時。

1. 資料模式

Lorel 所使用的資料模式稱為物件交換模式(Object Exchange Model, OEM)。在 OEM 模式中，資料以圖形(Graph)方式呈現，頂點(Vertexes)代表物件(Objects)，連接(Edges)代表資料本身(Labels)。所有實體(Entities)在 OEM 中皆為物件，每個物件都有其唯一的識別碼(Oid)，且物件可分為兩種類型 Atomic 與 Complex。

2. 查詢語言

Lorel 查詢語言是 OQL 的延伸，其查詢的型式是以路徑表達(Path Expressions)為基礎的導覽(Navigation)查詢，可分為兩種查詢語法的表達方式：(1)簡單路徑

表達(Simple Path Expressions)；(2)一般路徑表達(General Path Expressions)。

三、WebOQL

WebOQL 是由多倫多大學所提出[2]。WebOQL 對於三種資料來源，紀錄為主的資料(Record-Based Data)，結構化文件(Structured Documents)，與超文件(Hypertexts)，允許由三種中的任一種物件型態實例(Instances)，再建構為任何另一種的實例。

1. 資料模式

WebOQL 資料模式抽象化的程度不似 OEM 而是屬於輕量級的資料模式，亦不似傳統的重量級的綱要式資料模式，而是類似編譯器的概念，為了最佳化的中介語言。其基礎是有序樹(Ordered Trees)，把每個 Web 都當成是一個樹狀圖，而主要的資料結構稱為 Hypertrees。Hypertrees 為有序、由箭頭標籤(Arc-Label)組成的樹狀圖，其中，Arc 又可分為兩種型態，內部與外部。內部箭頭(Internal Arcs)代表結構化的物件，外部箭頭(External Arcs)代表一些參考的超連結物件。

2. 查詢語言

WebOQL 為完全的函式語言，其語法除了類似 select-from-where 子句，另外還包括一些運算子功能，像是 Prime、Peek、Hang、Concatenate、Head 與 Tail。

四、ARANEUS

ARANEUS 由義大利 Universita di Roma Tre 的資料庫團隊所開發，為一套語言的集合，對來自 WWW 的資料，做管理與再建構[3]。主要解決 Web 兩個問題：(1)超文件的結構鬆散且無法對 Web 資訊提供全面性的視野(Global View)；(2)對 Web 資料作運算需要花費額外的心力。利用傳統關連式資料庫的技術來對 Web 資料作重組與整合。其架構分成三個層面探討，(1)擷取半結構化的 Web 資料並轉換為資料庫結構化的 View；(2)以傳統資料庫技術來對此 View 做資料重組與整合；(3)資料整合後，重新推導出新的 Hypertext 文件。

1. 資料模式

ARANEUS的資料模式簡稱ADM，是物件資料模式的變形。每個Web網頁可以視為一個物件，包括了辨識碼與一連串的屬性。網頁綱要(Page Scheme)的概念類似關連式資料庫中的關連，塑模一連串的同質網頁。網頁的屬性可能為簡單或複雜型態，簡單型態的屬性像是文字(Text)，圖形/Images)，二元資料(Binary Data)或連接到其他的網頁；複雜型態屬性則像是項目條列等。

2. 查詢語言

基於 ARANEUS 的三層架構，它發展了三種查詢語言來搭配其 View 的定義處理：(1) EDITOR 看到網頁綱要的實例(Instances)，以編輯網頁文字內容而設計，功能是網頁 Wrappers；(2) ULIXES 對 Web 建立資料庫的 Views，以使用資料庫做資料分析與整合；(3) PENELOPE 允許從關連的 Views 重新定義並推導出新的 Hypertext 文件。

五、超文件資料模式與查詢語言之比較

本節我們共談論了四種主要半結構化文件查詢語言，依據[15]研究調查文獻分為四個維度來探討半結構化文件查詢語言的特色與異同，四個維度包括資料模式、語言格式、是否包含路徑表達、是否有圖形產生等，表一共集合九種半結構化文件查詢語言，對它們做一個比較分析。

表一 半結構化和超文件查詢語言的比較 資料來源：本研究

查詢語言	資料模式	語言格式	是否包含路徑表達	是否有圖形產生
WebSQL	Relational	SQL	是	否
W3QS	Labeled multigraphs	SQL	是	否
WebLog	Relational	Datalog	否	否
Lorel	Labeled graphs	OQL	是	否
WebOQL	Hypertrees	OQL	是	是
UnQL	Labeled graphs	Structural recursion	是	是
STRUDEL	Labeled graphs	Datalog	是	是
ARANEUS	Page schemes	SQL	是	是
FLORID	F-logic	Datalog	是	否

六、XML-QL

XML-QL 是由 AT&T Labs 與賓州大學合作開發[8]為解決 XML 文件中遇到上述問題的查詢語言。它是一項宣告性的、關連完整的查詢語言，可由現存的 XML 文件擷取資料並建構出另一份新的 XML 文件。

七、XQL

繼 XML-QL 後，W3C 於 1998 年 9 月又提出了 XQL，全名 XML Query Language [7]，係為查詢 XML 文件、XML 文件集合與 XML 網站而設計，能由 XML 文件中尋找與過濾出資料元素以及特定字元，藉由 XQL 搜尋 XML 檔案儲存庫，並建立其他應用程式該特定元素之超鏈結。設計皆來自於 XML-QL，在當時期望延續 SQL/OQL 成為資料庫的一項新標準。

八、Quilt

繼 XQL 之後，W3C 2000 年又提出了 Quilt 查詢語言[5]，其本身源自諸多的 XML-based 語言如 XQL, Xpath, XML-QL, YATL 以及 XSQL。這些查詢語言本身皆有其應用的狀況。Quilt 擷取上述各項查詢語言獨到之處，擷取 XQL 與 Xpath 對階層文件的導覽語法，利用 XML-QL 連結變數的概念來產生新的結構，結合 SQL 的 select-from-where 的基本查詢模式及引用 OQL 函式的概念，整合而成新的查詢語言。其中，XQL 以及 XML-QL 是影響 Quilt 的最主要查詢語言，因此 Quilt 的許多語法皆繼承自 XQL 以及 XML-QL。

九、Xquery

2001 年 7 月，W3C 的 XML Query Working Group 又依據 Quilt，衍生出 Xquery，成為目前 W3C 2003 年主要使用的 XML 查詢語言標準。其繼承之前其他查詢語言的許多特點。Xquery 基於 XML 綱要，擁有一型態系統，其所使用的型態皆定義於 <http://www.w3.org/2001/XMLSchema> 中的 namespace (稱為 xsd)。所有原始的或推導的資料型態皆可使用於查詢中。

十、Lorel

鑒於 XML 逐漸成為 WWW 上資料呈現與交換的新標準，研究者發現 XML 與半結構化資料有許多相似的地方，也有相異的地方。因此 Lorel 語言針對 XML 的特性，將其現有的 Lorel 資料模式與查詢語言做修改，以因應處理 XML 的資料[18]。

十一、XML 文件資料模式與查詢語言之比較

本節介紹五種 XML 文件的查詢語言，除了 Lorel 是由史丹佛大學所開發，其餘四種皆由 W3C 所提出，因此許多語法與特性都相互繼承，除了這兩個組織，開發 SQL 語言的 NCITS H2 and SC32 組織，也在 2000 年的時候，籌劃改良 SQL

語言使其適用於查詢 XML 文件，目前稱 SQL/XML[10]。表二對本節所介紹的 XML 查詢作分析比較。

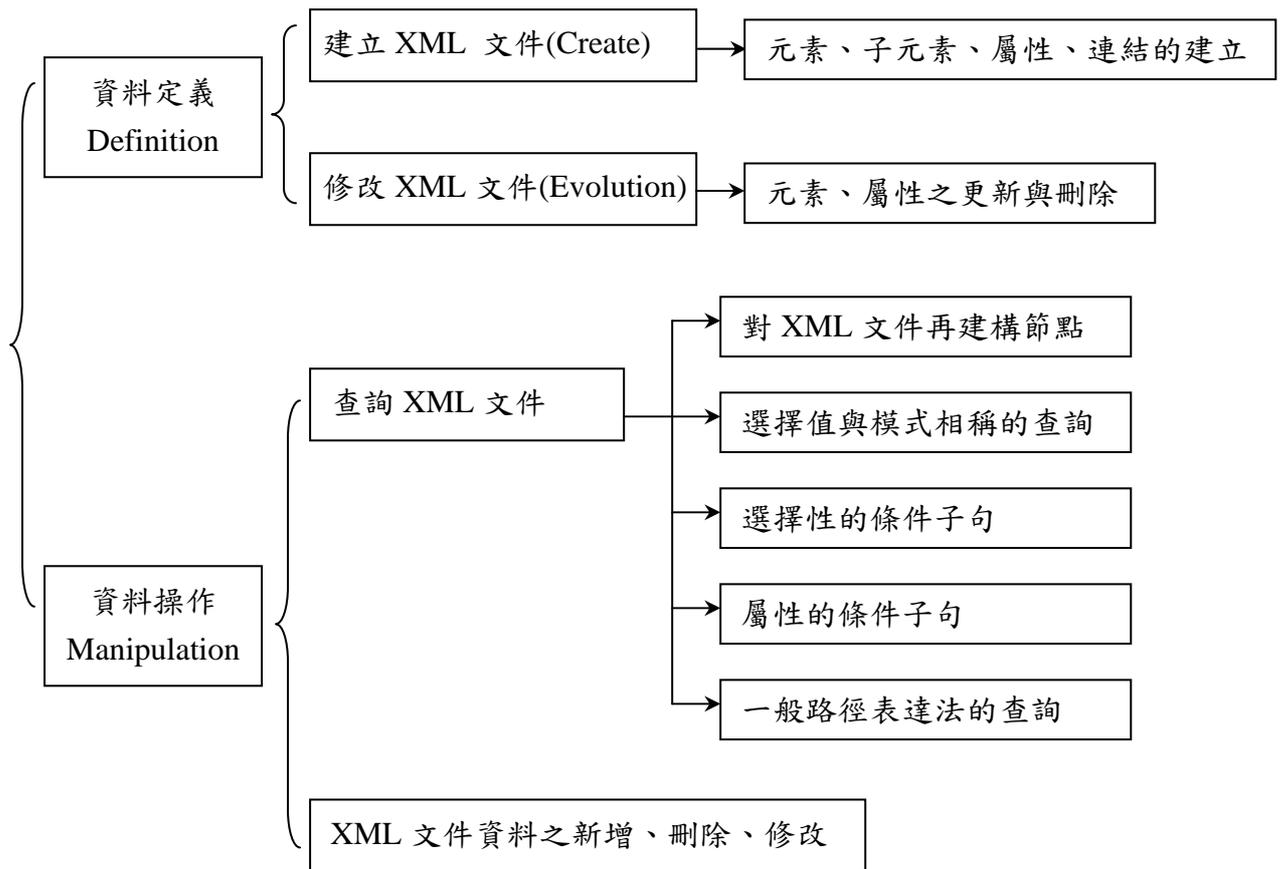
表二 XML 文件查詢語言的比較 資料來源：本研究

查詢語言	開發單位	時間	資料模式	特色
XML-QL	W3C	1998 年 8 月	樹狀圖(分有序與無序)	查詢利用元素模式 (Element Patterns)來對應資料
XQL	W3C	1998 年 9 月	無特別說明，主要利用 DTD 表達資料綱要	延伸自 XML-QL，語法類似 SQL、OQL，以 select-form-where 為基本型態
Quilt	W3C	2000 年 6 月	樹狀圖(節點可區分為元素，屬性，文字等三種)	主要繼承自 XML-QL 與 XQL，查詢稱為 FLWR 表達法，語法以精簡為主
Xquery	W3C	2001 年 7 月 2003 年	同 Quilt，但節點為有序排列	依據 Quilt 衍生而成，特色為允許一些使用者的自訂資料型態
Lorel	Stanford University	1999 年 6 月	改良的 OEM 圖形模式	不只為查詢語言，也為資料更新語言
SQL/XML	NCITS H2 and SC32	2000		

參、研究方法

一、XML 文件資料模式至關連資料模式轉換

圖一為雙向由 XML 文件資料模式為首轉換和互動為關連式資料表的資料模式的研究架構示意圖。



圖一 XML 文件轉換至關連資料表的轉換架構 資料來源：本研究

我們分為資料定義與資料操作兩個方向研發，資料定義部分探討當前端來源為 XML 文件時，後端關連資料表是如何的建立與修改是具複雜性和專屬性。建立 XML 文件包括建立 XML 的元素、子元素、屬性與連結時轉換為在關連資料表中是如何的建立。修改 XML 文件則說明當前端 XML 元素、子元素、屬性、連結有修改異動時，在後端關連資料表中如何同步的修改。資料操作部分探討查詢 XML 文件，及 XML 文件資料之新增、刪除、修改。查詢 XML 文件探討資料轉換為關連資料表後，如何將對前端 XML 文件的主要查詢類型，轉換為對資料表作 SQL 查詢，且其語法的表達為何具複雜性和專屬性。最後 XML 文件資料之新增、刪除、修改則探討前端 XML 資料內容的修改，在關連資料表的做法。

1.建立 XML 文件

本研究所引用的 XML 資料模式是依據 Florescu 和 W3C[15] [16]所提出的標準的 OEM 和 Object Model 為簡單圖形，其中節點代表物件，有其 oid，連結代表屬性名。每個內部物件可能有一連串的屬性，皆用連接表示，葉子則代表資料值。若連結的目標物件為葉子，我們稱為資料屬性，若否，我們稱為參考屬性。

2.修改 XML 文件

修改 XML 文件包括對元素、屬性的更新與刪除。

(1)元素的更新

當欲新增一元素時，只需對該元素的每個屬性，資料表中增加一筆記錄。

(2)屬性的更新

當欲新增一屬性時，只需對該的於資料表中增加一筆記錄，但特別注意的是，修改 Source 為同元素其 Order 欄位的值，若新增的屬性為新的屬性，我們要新增此一屬性名的資料表。

(3)元素的刪除

XML 的資料模式若欲刪除一元素時，對應至關連表格做法則是將其相關的屬性記錄皆刪除。

(4)屬性的刪除

若 XML 的資料模式若欲刪除一屬性連結時，至該屬性表格，直接找出欲刪除的 Source 元素，並將該筆記錄刪除，唯一要注意的是，相同 Source 的屬性其 Order 的順序資料要順應作修改調整。

3.查詢 XML 文件

關連建立完成後，對 XML 的查詢，轉成 SQL 查詢，Xquery 的語法轉換皆能對應至關聯資料庫 SQL 語法中的運算，我們舉出五種主要查詢 XML 文件的查詢類型，當資料來源轉換為關連式表格時，其查詢語法的轉變。

(1)對 XML 文件再建構節點 (Reconstructing Objects)：此類查詢是指給定一物件的 Oid，並傳回其所有的屬性值。

(2)選擇值(Selections Value)的查詢：選擇值查詢是指傳回符合某項條件的 Oid。

(3)選擇性的條件子句(Optional Predicates)：對於不規則的 XML 資料，Xquery 亦提供選擇性的條件子句的查詢。

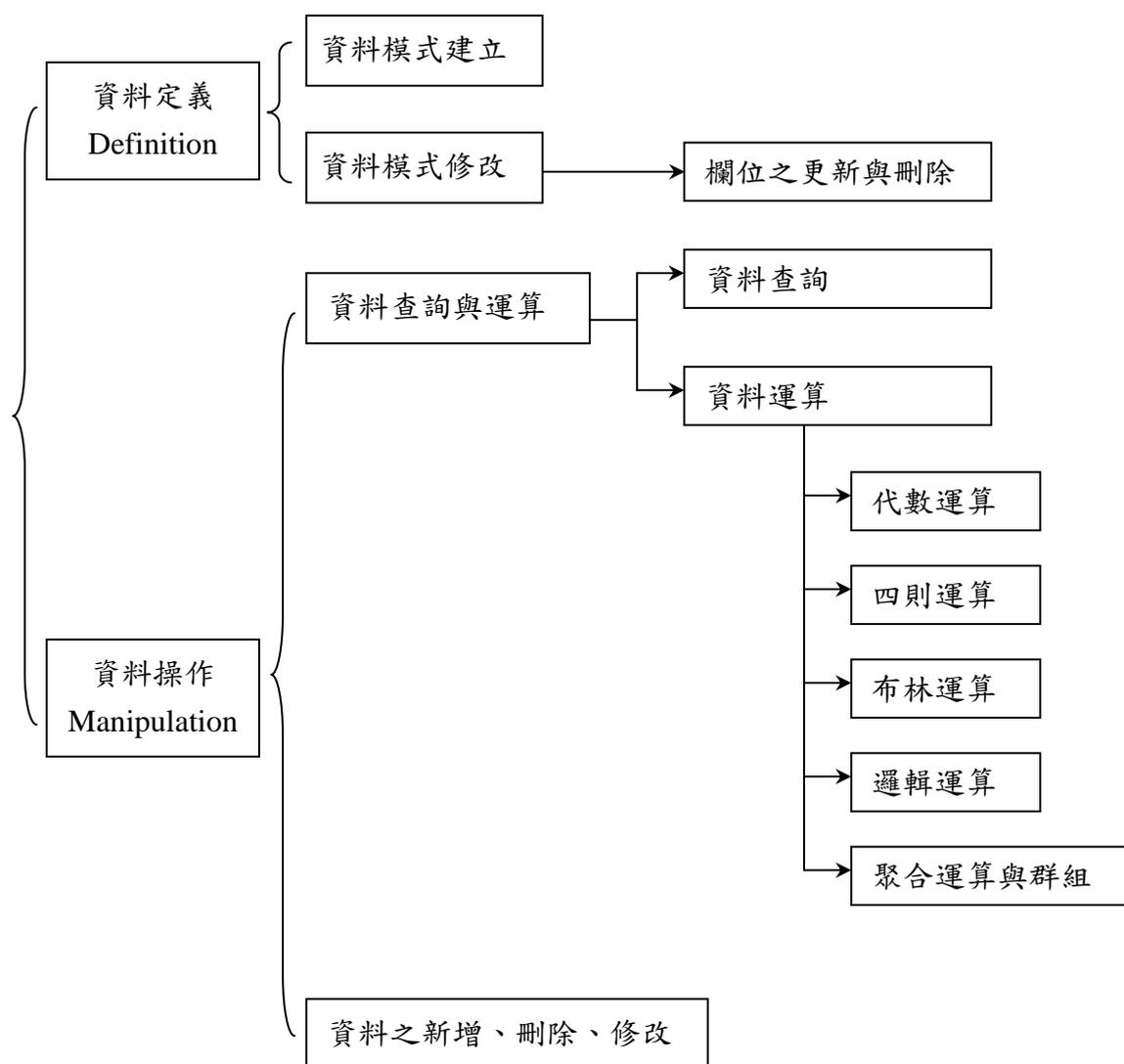
(4)屬性(Predicates on Attributes)的條件子句：Xquery 的另一項特色是支援對屬性名的條件查詢。

(5)一般路徑表達法(Regular Path Expressions)的查詢：此類查詢是指找出參考到一特殊 a_1 值所連接的所有物件。

(6)當前端 XML 文件資料內容有做新增、刪除、修改時，在後端資料表的部分，需要至更新的屬性表格將其修改的記錄值作同步修改。

二、 關連資料模式至 XML 文件轉換

圖二為雙向由關連式資料表的資料模式為首轉換和互動為 XML 文件資料模式的研究架構示意圖。



圖二 關連式的資料表轉成 XML 文件轉換架構說明 資料來源：本研究

分為資料定義與資料操作兩個方向研發，資料定義部分探討當前端來源為關連資料表時，後端 XML 資料模式是如何的建立與修改，資料模式建立包括資料表綱要的建立。資料模式修改則說明當前端關連資料表綱要有修改異動時，後端 XML 資料模式如何同步的修改具複雜性和專屬性。資料操作探討資料查詢與運算，及資料之新增、刪除、修改。資料查詢與運算探討資料轉換為 XML 資料模式後，如何將對前端關連資料表的查詢，轉換為對 XML 資料模式查詢，且其語法的表達為何具複雜性和專屬性。資料運算探討在關連資料表中，其查詢具備代

數運算轉至 XML 文件為何。最後資料之新增、刪除、修改則探討前端關連資料表的更新，在 XML 資料模式為何

1. 資料模式建立

(1) 單一表格轉換

單一表格其資料模式為資料表的綱要，若轉換為 XML 資料模式，資料表名稱為根節點，以主鍵為識別碼，以子樹方式呈現，根節點連接素集合，包含資料表中紀錄集合。

(2) 多重表格轉換

以兩個具參考關連的表格其在關連資料庫中的資料模式為實體關係圖與資料表綱要，資料表間是透過主鍵和外鍵作連結。將之轉換為 XML 資料模式，根節點為資料庫名稱，第一層的各連結為各資料表名稱，其後接各筆記錄當作各個元素，為樹狀圖的第二層，每個元素後，連接表格中的各欄位當作其子元素，為第三層。

2. 資料模式修改

區分為對欄位的更新、刪除及資料型態修改三部份來探討：

(1) 新增欄位

針對關連式資料庫的資料表資料模式(綱要)若欲新增欄位時，對應至 XML 資料模式，由欲修改的資料表節點開始，對其下所連接的所有元素(記錄)，皆增加一欲新增的子元素連結，增加欄位的動作。

(2) 刪除欄位

關連式資料庫的資料表資料模式(綱要)若欲刪除某欄位時，對應至 XML 資料模式，由欲修改的資料表節點開始，對其下所連接的所有元素(記錄)，刪除該項對應的子元素連結，刪除欄位的動作。

(3) 修改欄位名稱

關連式資料庫的資料表資料模式(綱要)若欲修改某欄位名稱時，對應至 XML 資料模式，對應的子元素名稱，皆更正為新的欄位名稱。

3. 資料查詢與運算

XML 資料模式建立完成後，對關連資料表的查詢，轉換和互動成 XML 文件的查詢與運算：

(1)資料查詢

關連式資料表轉為 XML 資料模式後，欲對資料作查詢，透過 XML 查詢語言常用的路徑表達法直接對 XML 模式作查詢。

(2)資料運算

相對於關聯式資料庫，將資料由關連資料模式轉為 XML 資料模式後，其查詢包含算數運算以對資料進行運算與分析，而轉換至 XML 文件後，仍然能夠適用的運算語法。

A.代數運算

選取(Restrict)、提取(Project)、相乘(Product)、相除(Divide)：由於 XML 資料模式為樹狀結構，不像關連資料模式為表格的二維關係，所以沒有提供選擇、擷取、乘積、分割等代數運算。

聯集(Union)、交集(Intersect)、差集(Difference)：然而 Xquery 中也提供這三種運算子，但不似關連模式以欄位當運算元，其運算元是節點，作用是合併節點序列以形成新的節點序列。其中 Union (“ | ”) 表傳回的運算元不是左邊的節點就是右邊的，Intersection 表傳回的運算元包括左邊與右邊的所有運算元，Except 表傳回的運算元節點只包含左邊的運算元，不包含右邊的。聯合(Join)：為查詢語言的重要運算，在 Xquery 中，其代表的是對多份 XML 文件來源的聯合。

B.四則運算

Xquery 單項算數運算子(+、-)，及二項算數運算子(+、-、*、/)，然而其相對的運算元為節點，又由於 XML 文件的資料模式皆為字串型態，所以做四則運算，先利用轉換函數將字串型態轉為數值或日期型態。

C.布林運算

在 Xquery 中只包括 AND、OR 兩個布林運算子，不支援 NOT 運算子。

D.邏輯運算

比較運算其中 =，!=，<，<=，> 與 >=等運算子是用來作相同型態的值的比較。而 == 與 != 則是用來對節點辨識碼作比較，比較兩個運算元是否為同一節點，邏輯運算傳回的結果通常為 True 或 False 的布林值。

E.聚合函數(Aggregation)與群組

在 XML 資料模式中的查詢，Xquery 包含聚合函數配合 GROUP BY 行群組化查詢的功能。

4.資料之新增、刪除、修改

(1)資料更新

在關連資料表中，若欲新增一筆記錄時，對應至 XML 資料模式，作法為資料表的根節點下，增加一元素的子樹，其中子樹的連接包含了各個欄位屬性，葉子則為其屬性值。

(2)資料刪除

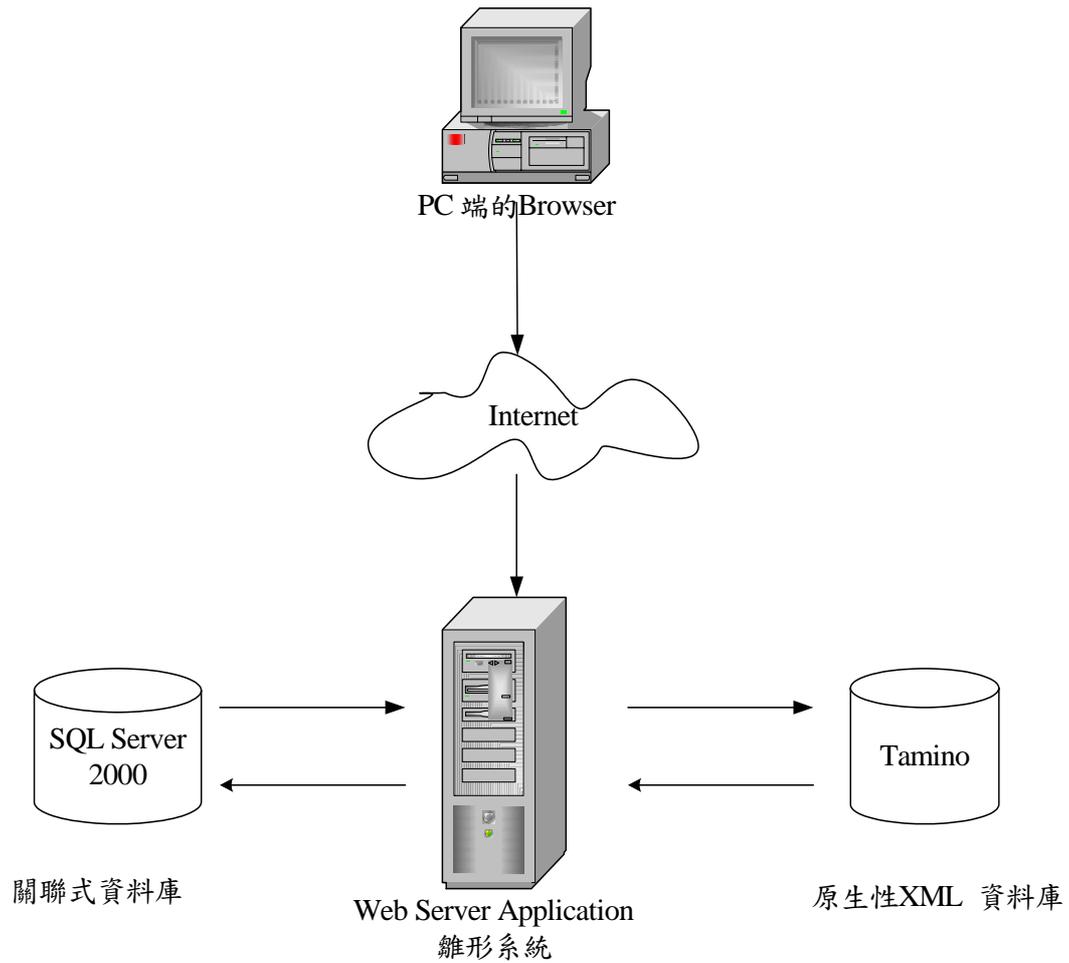
相同於新增資料，欲刪除資料表中的一筆記錄，對應至 XML 資料模式，將該筆記錄對應的元素於資料模式中刪除。

(3)資料修改

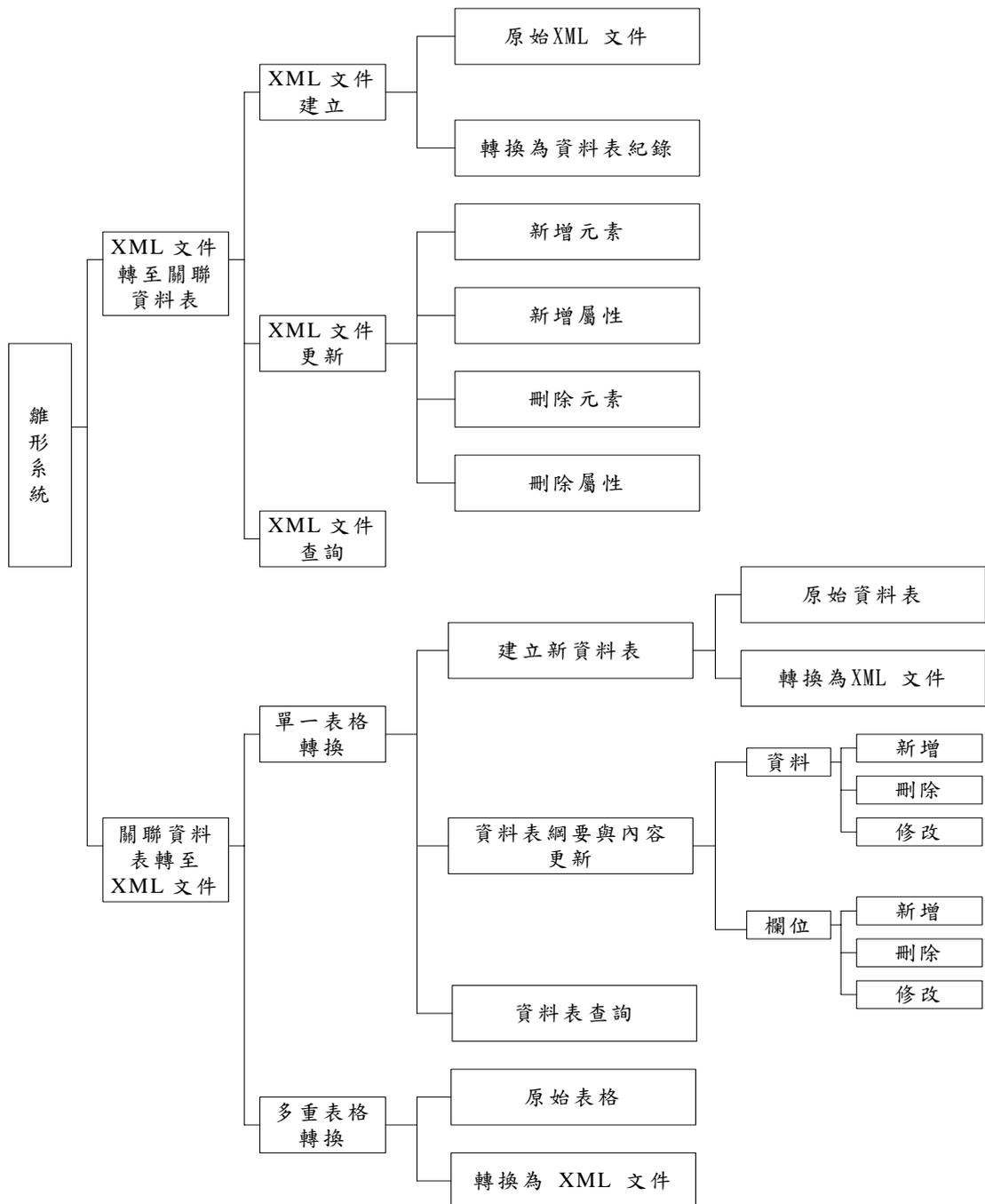
在關連資料表中，若欲修改某項資料內容時，對應至 XML 資料模式，作法為至 XML 資料模式中，欲修改元素的欄位屬性，將其葉子的值更改。

肆、研究實作

針對本研究方法，以關聯式資料庫與原生性 XML 資料庫為基礎，我們實作關聯式資料表與 XML 文件資料模式雙向和即時轉換和互動的雛形系統架構如圖三所示。由使用者端的瀏覽器，透過 Internet 連接至架構在 Web Server，透過雛形系統接收異質資訊來源進行處理儲存於目的資料庫中。雛形系統功能設計之架構圖如圖四所示。



圖三 雛形系統架構圖 資料來源：本研究



圖四 雛形系統設計架構圖 資料來源：本研究

一、XML 文件轉至關聯資料表

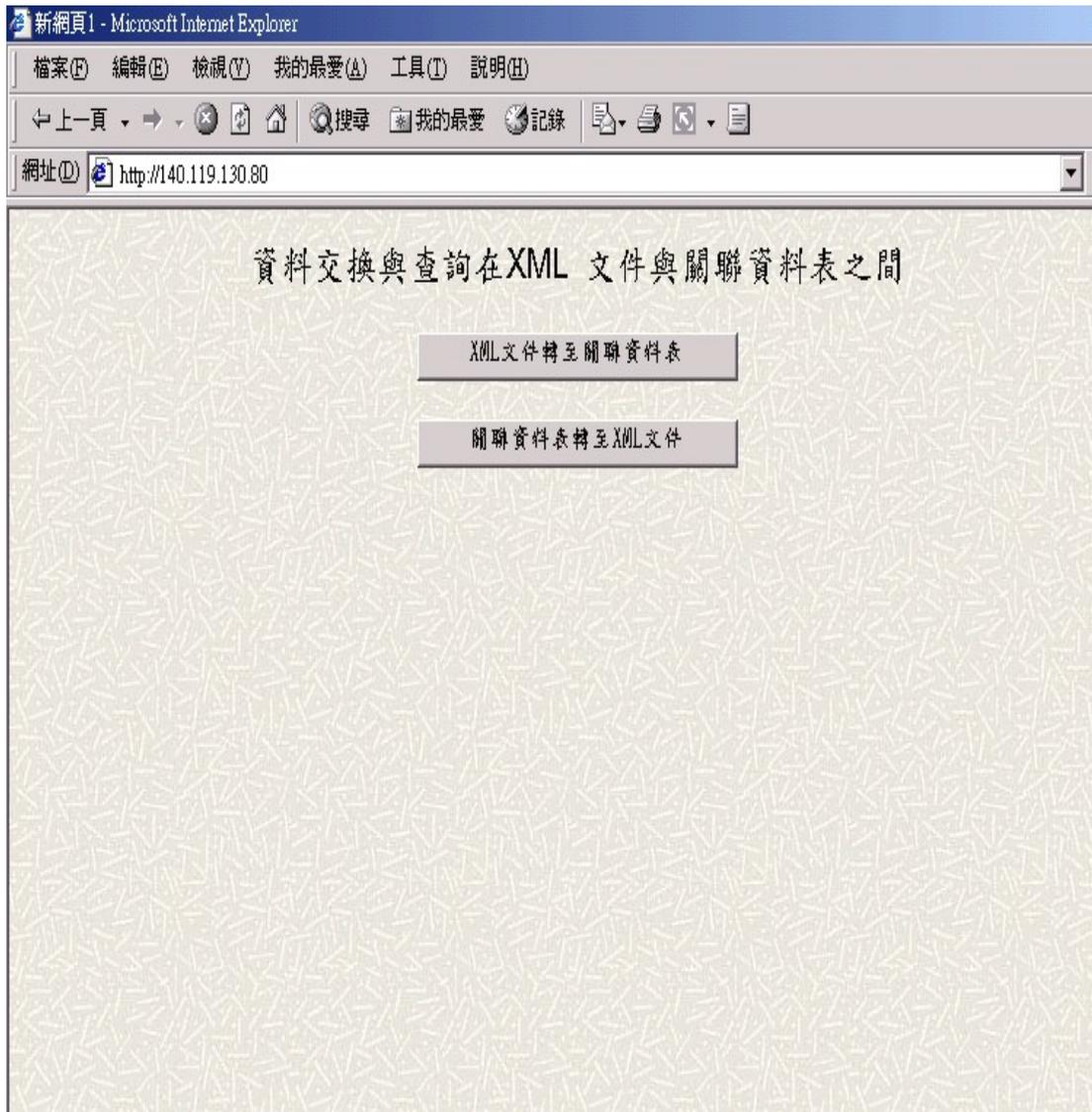
建立一份 XML 文件將其轉為關聯式資料表的方式存入關聯式資料庫。針對建立之 XML 文件，欲進行元素、屬性之更新時，對應至資料表，也同時進行資料之更新。針對簡單的 Xquery 查詢，對應轉換為 SQL 查詢並對資料表進行查詢後，再將結果以 XML 文件的形式傳回。

二、關聯資料表轉至 XML 文件

針對單一表格轉換，建立或選擇一資料表並利用研究方法，將其轉為一份 XML 文件。針對建立或選擇之資料表，欲進行資料與欄位之更新時，對應至 XML 文件，也同時進行資料與元素名稱之更新。將產生之 XML 文件存入 Tamino XML 資料庫，針對簡單 SQL 查詢，對應轉換為 Xquery 語法並對 XML 文件進行查詢後，再將結果以表格的形式傳回。針對多重表格轉換，兩個具參考關連的表格其在關連資料庫中的資料模式為實體關係圖與資料表綱要，資料表間是透過主鍵和外鍵作連結。本研究將之轉換為 XML 資料模式，根節點為資料庫名稱，第一層的各連結為各資料表名稱，其後接各筆記錄當作各個元素，為樹狀圖的第二層，每個元素後，連接表格中的各欄位當作其子元素，為第三層。表格間有利用外鍵引用到別的表格資料記錄的，再透過連結線作結合，以為其參考屬性。

三、簡易雛形系統

本節說明簡易雛形系統的實作範例。圖五為本系統的進入畫面，選擇要進入 XML 文件轉至關聯資料表子系統或是關聯資料表轉至 XML 文件子系統。



圖五 本系統的進入畫面 資料來源：本研究

圖六、圖七、圖八分別為 XML 文件轉至關聯資料表子系統的進入畫面、XML 原始文件、和轉換結果，輸入欲建立 XML 文件的新增元素或是選擇各項功能選項，簡易範例之原始文件和轉換互動後之資料表結果畫面。圖九、圖十、圖十一為關聯資料表轉至 XML 文件子系統的進入畫面、資料表原始資料、轉換為 XML 文件結果，。

XML 文件轉至關聯資料表

新增元素

請輸入元素 id:

請輸入元素名稱:

請輸入屬性順位:

請輸入屬性名稱:

屬性為資料屬性?

是 請輸入資料

否 請輸入參考屬性之 id

傳送

查詢

回首頁

觀看原始XML文件

觀看轉換至資料表紀錄

繼續新增元素

新增屬性

刪除元素

刪除屬性

圖六 XML 文件轉至關聯資料表子系統的進入畫面 資料來源：本研究

```
<?xml version="1.0" encoding="Big5" ?>
- <profile>
  - <person id="1">
    <age>55</age>
    <name>peter</name>
    <address>4711 Fruitdale Ave.</address>
    <child>3</child>
  </person>
  - <person id="2">
    <age>38</age>
    <name>Mary</name>
    <address>4711 Fruitdale Ave.</address>
    <child>4</child>
    <hobby>painting</hobby>
  </person>
  - <person id="3">
    <age>22</age>
    <name>John</name>
    <address>5361 Columbia Ave.</address>
    <hobby>swimming</hobby>
  </person>
</profile>
```

圖七 原始 XML 文件 資料來源：本研究

name

Source	Ord	Valint	Valstring	Target
1	2		peter	
2	2		Mary	
3	2		John	

age

Source	Ord	Valint	Valstring	Target
1	1	55		
2	1	38		
3	1	22		

address

Source	Ord	Valint	Valstring	Target
1	3		4711 Fruitdale Ave.	
2	3		4711 Fruitdale Ave.	
3	3		5361 Columbia Ave.	

child

Source	Ord	Valint	Valstring	Target
1	4			3
1	5			4
2	4			4

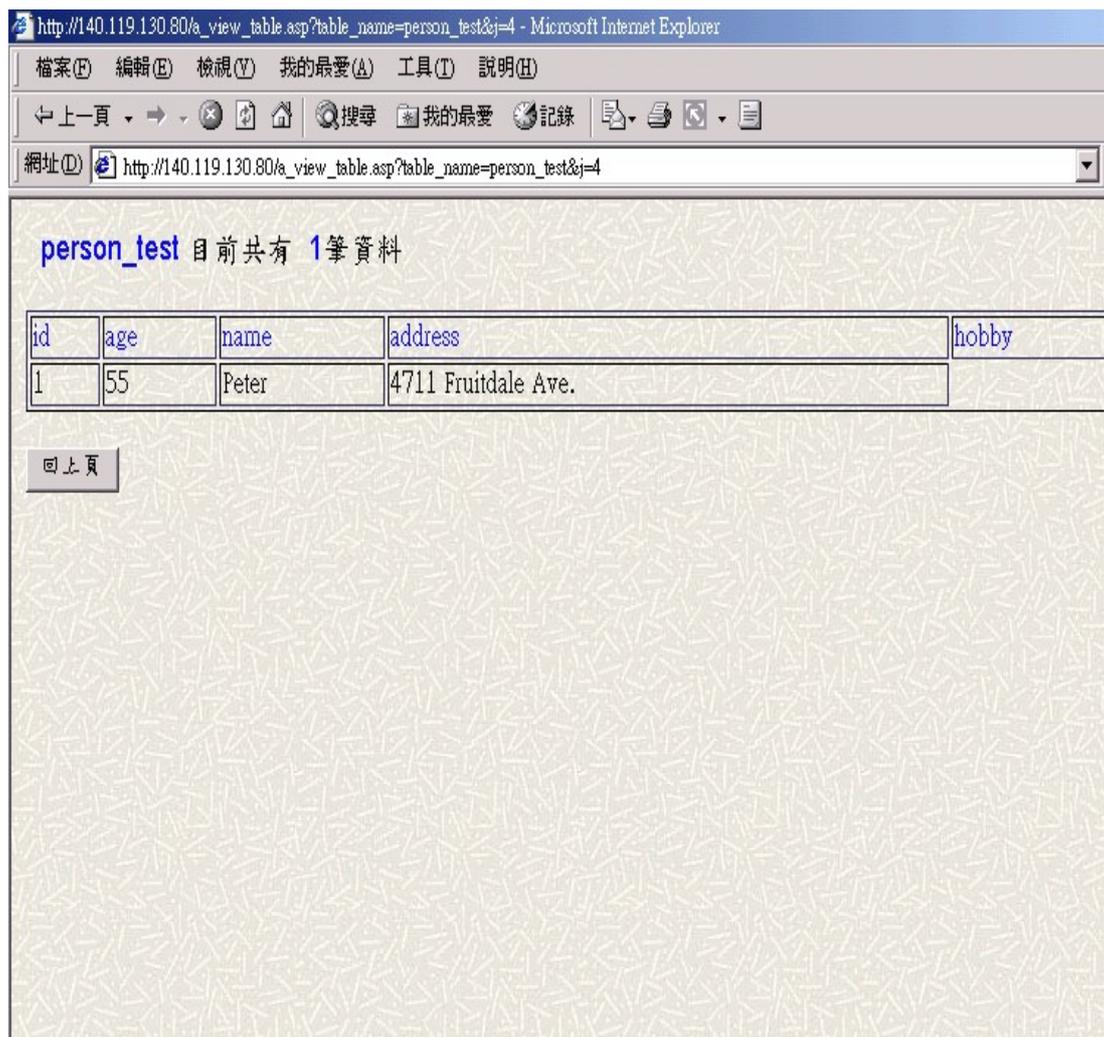
hobby

Source	Ord	Valint	Valstring	Target
2	6		painting	
3	4		swimming	

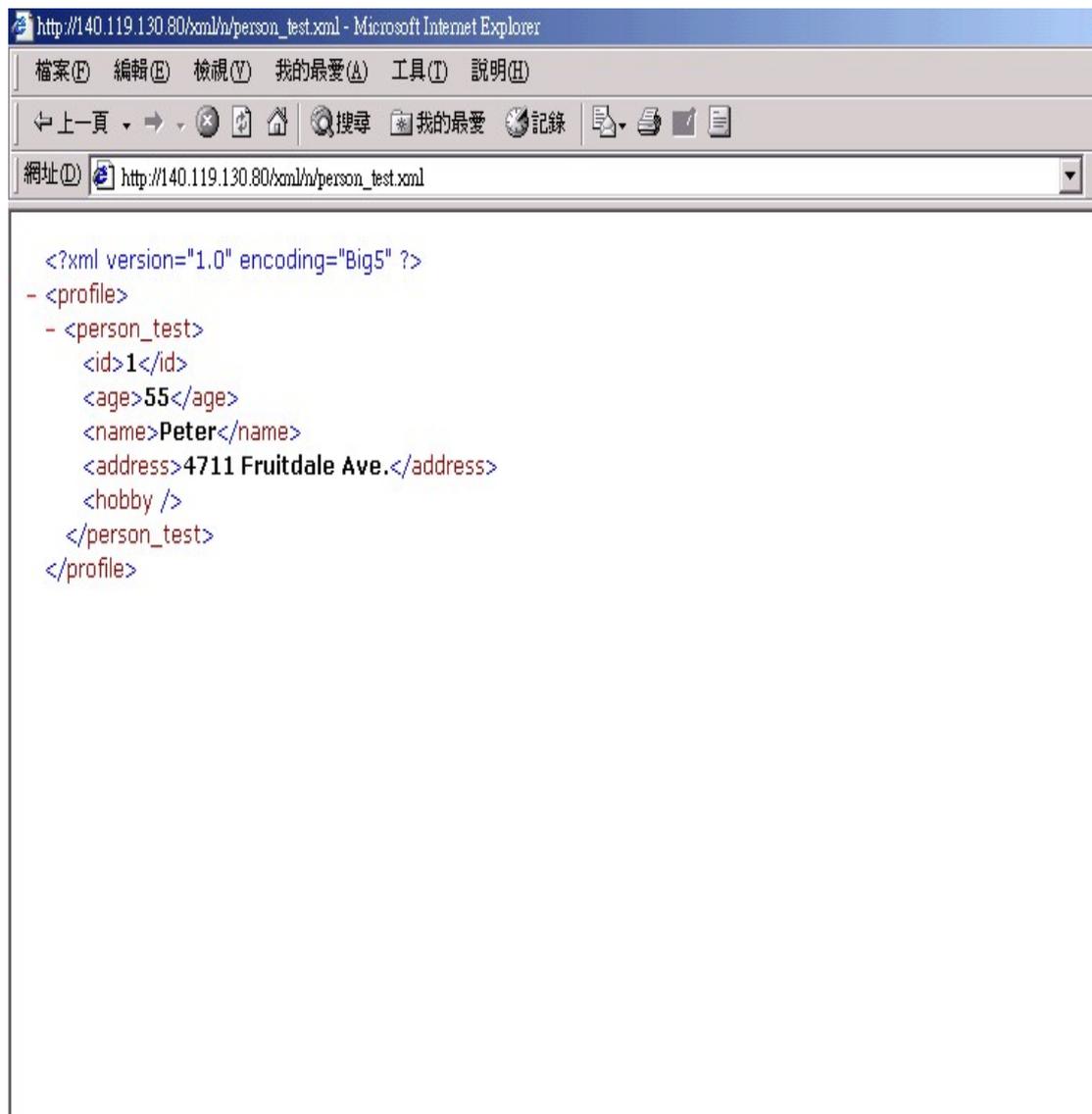
圖八 轉換至資料表結果畫面 資料來源：本研究



圖九 關聯資料表轉至 XML 文件子系統的進入畫面 資料來源：本研究



圖十 原始資料表 資料來源：本研究



圖十一 轉換為 XML 文件結果 資料來源：本研究

伍、研究討論

網際網路上異質資料的整合，及 XML 文件與傳統關聯式資料庫資料交換與轉換及互動的課題，讓異質資料可雙向轉換並同步更新，是本研究的研究動機。對於 XML 文件與關聯式資料轉換的問題，目前各解決方案將 XML 文件儲存於關聯式資料庫中，非真實轉換為純表格形式，以表格的形式儲存 XML 文件，其在綱要的結構上失真。本研究試圖在方法與技術上提出轉換模式，對異質性 XML 文件與關聯式資料作雙向之轉換並達到同步更新與異質查詢的能力，增加轉換資料之精確性。

一、XML 文件轉為關聯資料表部分

績效評估法則之 Attribute + Inlining 方法，最短時間與最節省資料庫空間，將 XML 文件完整的以表格方式儲存於關聯式資料庫中，並經由關聯資料庫的處理，對原始 XML 文件達到元素與屬性的更新動作，並將結果以 XML 文件的形式傳回，達到資料的透通性的處理與轉換。

二、關聯資料表轉為 XML 文件部分

以欄位為子元素名稱，各別資料錄為元素資料，將關聯式資料表以 XML 文件的形式增加彈性，關聯式資料透過網路作資料交換與傳輸的動作，省去人工轉換格式，對於資料表更新維護對應 XML 文件同時完成。

三、查詢語言的轉換部分

對於關聯式資料庫與 XML 文件其在查詢語言上，皆有各自的查詢語言標準，分別為 SQL 語言與 Xquery 語言。本研究之資料轉換模式，對於查詢語言之轉換上，解析兩者之查詢語法，是同步且對應的資料的雙向轉換。

四、讓關聯式資料庫與 XML 文件之整合應用

本研究雛形系統之關聯式資料庫表格與 XML 文件資料整合與轉換，關聯式資料表可由 XML 文件之形式，直接在網路上傳輸，整合至不同企業之中，無需任何加工的轉換處理。企業接收 XML 文件，透過本系統轉換儲存於關聯資料表中，運用既有關聯資料庫之強大運算分析能力，對 XML 文件資料進行分析處理。

五、提供異質資料來源資料即時的同步更新與查詢

針對異質資料來源之資料綱要與資料有系統且雙向同步的維護與更新，並配合查詢語言之轉換，是目前研究尚無一套如此完整之解決方案，本研究對其提出改進。

六、對異質資料來源轉換較低的建置與維護成本

XML 文件與關聯式資料表在異質資料的轉換上，減低人工方式處理資料轉換的工作所耗費之時間與金錢，且在異質資料的更新維護上由雛形系統展示。

陸、結語與未來研究方向

一、結語

1. 關聯式資料庫與 XM 文件之整合應用

本研究之模式和方法，讓關聯式資料庫表格與 XML 文件達到資料整合與轉換，讓關聯式資料表由 XML 文件之形式，直接在網路上傳輸，整合至不同運算和流程之中，無需人工轉換處理。XML 文件透過關聯資料表，運用既有關聯資料庫之強大運算分析能力，對 XML 文件資料進行分析處理。

2. 異質資料來源資料即時的同步更新與查詢

針對異質資料來源之資料綱要與資料有系統且雙向同步的維護與更新，並配合查詢語言之轉換提出完整之解決方案。

3. 異質資料來源轉換較低的建置與維護成本

透過本研究之雛形系統，對 XML 文件與關聯式資料表在異質資料的轉換上，減低資料轉換的工作所耗費之時間與成本。

二、未來研究方向

本研究之模式和方法與雛形系統，在未來尚有進一步探討的空間。首先考慮動態產生之關聯式資料表轉換成的 XML 文件之前，動態產生符合 Tamino XML DBMS 規定標準之對應 XML 文件綱要，以便透過系統直接將 XML 文件綱要與資料存入 Tamino 資料庫中，如此便可透過系統對 XML 文件進行查詢。接著，本研究產生之 XML 文件尚屬於原始之 XML 文件，未搭配任何的 XSLT 以對 XML 文件在瀏覽器或是其他之輸出設備如 WAP、PDA、Tablet PC 等資料之呈現，皆將對這些議題作進一步之研究探討。

柒、參考文獻

1. Abiteboul, Serge, Dallan Quass, Jason McHugh, Jennifer Widom and Janet Wiener (1997), "The Lorel Query Language for Semistructured Data," *International Journal on Digital Libraries*, 1(1), pp68-88.
2. Arocena, Gustavo O. and Alberto O. Mendelzon (1998), "WebOQL: Restructuring Documents, Databases and Webs," *In Proceedings of International Conference on Data Engineering (ICDE)*.
3. Atzeni, Paolo, Giansalvatore Mecca and Paolo Merialdo (1997), "To Weave the Web," *In Proceedings of the International Conference on Very Large Data Bases*, pp. 206-15.
4. Buneman, Peter, Susan Davidson, Gerd Hillebrand and Dan Suciu (1996), "A Query Language and Optimization Techniques for Unstructured Data," *In Proceedings of ACM SIGMOD Conference on Management of Data*, pp. 505-16.
5. Chamberlin, Don, Jonathan Robie and Daniela Florescu (2000), "Quilt: An XML Query Language for Heterogeneous Data Sources," *Proceedings of WebDB 2000 Conference*.
6. Chamberlin, Donald D. (2001), "Query Languages and XML," *IDEAS 2001*, pp. 297-300.
7. Decker, S. et al. (2000), "The Semantic Web: The Roles of XML and RDF," *IEEE Internet Computing*, Sep/Oct, 63-74.
8. Deutsch, A., Mary Fernandez, Daniela Florescu, Alon Levy and Dan Suciu (1999), "A Query Language for XML," *Computer Networks*, 31, pp.1155-69.
9. DeWitt, David, Jeffrey Naughton, Jayavel Shanmugasundaram, Kristin Tufte, Gang He and Chun Zhang (1999), "Relational Databases for Querying XML Documents: Limitations and Opportunities," *In Proceedings of the 25th VLDB Conference*, pp.25-32.
10. Eisenberg, Andrew and Jim Melton (2001), "SQL/XML and the SQLX Informal Group of Companies," *ACM SIGMOD RECORD*, 30(3), pp.105-8
11. Fankhauser, Peter (2001), "Xquery Formal Semantics: State and Challenges," *ACM SIGMOD RECORD*, 30(3), pp.14-9.
12. Fankhauser P., M. Marchiori and J. Robie (2000), "XML Query Requirements," <http://www.w3.org/TR/xmlquery-req>
13. Fensel, D. et al. (2001), "OIL: An Ontology Infrastructure for the Semantic Web," *IEEE Intelligent Systems*, 16(2), Mar./Apr., 38-45.
14. Fernandez, Mary, Daniela Florescu, Jaewoo Kang, Alon Levy and Dan Suciu (1998), "Catching the Boat with Strudel: Experiences with a Web-Site Management System," *In Proceedings of ACM SIGMOD Conference on*

Management of Data.

15. Florescu, D. and D. Kossmann (1999), "Storing and Querying XML Data Using an RDBMS," *IEEE Data Engineering Bulletin*, 22(3), pp.27-34.
16. Florescu, Daniela, Alon Levy and Alberto Mendelzon (1998), "Database Techniques for the World Wide Web: A Survey," *ACM SIGMOD RECORD*, pp.59-74.
17. Florescu, Daniela, Alon Levy, Mary Fernandez and Dan Suciu (1997), "A Query Language for a Web-Site Management System," *ACM SIGMOD RECORD*, 26(3), pp.4-11.
18. Goldman, R., J. McHugh and J. Widom (1999), "From Semistructured Data to XML: Migrating the Lore Data Model and Query Language," Proceedings of the 2nd International Workshop on the Web and Databases (WebDB '99).
19. Heflin, J. and J. Hendler (2001), "A Portrait of the Semantic Web in Action," *IEEE Intelligent Systems*, 16(2) 54-59.
20. Himmeröder, Rainer, Georg Lausen, Bertram Ludäscher and Christian Schleppehorst (1997), "On a Declarative Semantics for Web Queries," In Proceedings of the International Conference on Deductive and Object-Oriented Databases, pp. 386-98.
21. Klein, M. (2001), "XML, RDF, and Relatives," *IEEE Internet Computing*, March/April, 26-28.
22. Konopnicki, David (1995), "W3QS: A Query System for the World Wide Web," In Proceedings of the International Conference of Very Large Data Bases, pp54-65.
23. Kumar, Anil and Palvia Prashant (2001), "Key Data management Issues in a Global Executive Information System," *Industrial Management and Data Systems*, Vol. 101, No. 4, pp. 153-164.
24. Lakshmanan, Laks, Fereidoon Sadri and Iyer N. Subramanian (1996), "A Declarative Language for Querying and Restructuring the Web," In Proceedings of 6th International Workshop on Research Issues in Data Engineering, RIDE '96.
25. Melton, J. (1999), Complete Guide to SQL2 and SQL3, Ed. Morgan Kaufmann Inc., 1999.
26. Mendelzon, Alberto O., George A. Mihaila and Tova Milo (1996), "Querying the World Wide Web," In Proceedings of PDIS '96.
27. Naughton, Jeffrey, David DeWitt, David Maier and many others (2001), "The Niagara Internet Query System," *IEEE Data Engineering Bulletin*, 24(2), pp. 27-33.
28. Shanmugasundaram, Shekita, Kiernan, Krishnamurthy, Viglas, Naughton and Tatarinov (2001), "A General Technique for Querying XML Documents Using a

Relational Database System,” *ACM SIGMOD RECORD*.

29. Zhang, Qingyu (2001), “Object-Oriented Database Systems in Manufacturing: Selection and Applications,” *Industrial Management and Data Systems*, Vol. 101, No. 3, pp. 97-105.
30. W3QS: <http://www.computer.org/proceedings/icde/7807/78070586abs.htm>
31. XML-QL:<http://www.w3.org/TR/NOTE-xml-ql/>
32. XQL: <http://www.w3.org/TandS/QL/QL98/pp/xql.html>
33. Xquery: <http://www.w3.org/TR/2001/WD-xquery-20010607/>
34. PCMAG: <http://www.pcmag.com>
35. WWW: <http://w3.org>
36. OASIS: <http://www.oasis.org>
37. XMLDB: <http://www.xmldb.org>
38. XMLDB: <http://www.xmldb.com>