行政院國家科學委員會補助專題研究計畫成果報告

※※※※※※※※※※※※※※※※※※※※※※※※※
※　　　　　　　　　　　　　　　　　　　　　　　※
※　**分散式數位圖書資訊系統整合架構 (DL@NCTU) 之研究**　※
※　　　　　　　　　　　　　　　　　　　　　　　※
※※※※※※※※※※※※※※※※※※※※※※※※※

計畫類別：□個別型計畫　　■整合型計畫

計畫編號：NSC89－2218－E－009－009－

執行期間： 89 **年** 8 **月** 1 **日至** 90 **年** 7 **月** 31 **日**

計畫主持人：柯皓仁

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

執行單位：國立交通大學圖書館

中　華　民　國 90　**年**　10　**月** 25 **日**

分散式數位圖書資訊系統整合架構之理論研究與實作

The Research and Implementation on an Integrated Architectures of Distributed Digital Library Information Systems

計畫編號：NSC 89-2218-E-009-009
執行期限：89 年 8 月 1 日至 90 年 7 月 31 日

主持人：柯皓仁　交通大學圖書館
計畫參與人員：黃夙賢　交通大學資訊科學系
計畫參與人員：曾志軒　交通大學資訊科學系
計畫參與人員：嚴文亨　交通大學圖書館

一、中文摘要

在分散式的環境當中，數位圖書資訊的整合必須考慮資訊系統的異質性以及整合架構的共通性兩點因素。在本計劃中，我們利用詮釋資料描述語言(MML)來表示數位圖書資訊系統異質性的詮釋資料。透過詮釋資料描述語言當中的轉換機制，可以將異質性的詮釋資料轉換成專屬的資料格式。並且提出了虛擬聯合目錄的架構(VUCS@NCTU)，利用結構化文件當中具有共同結構的特性，擷取各數位圖書資訊系統當中的網頁資料，並且將擷取後的資料包裝誠詮釋資料描述語言的格式回傳給虛擬聯合目錄伺服器整合。我們並以各校圖書館館藏系統為例，實作虛擬聯合目錄館藏查詢系統來驗證本計劃中提出的分散式數位圖書資訊系統整合架構的可行性。

關鍵詞：分散式環境、數位圖書館、詮釋資料描述語言、虛擬聯合目錄

Abstract

In the distributed environment, successfully integrating digital libraries must consider two key issues: diversity of digital library system and interoperability of integration infrastructure. In this project, we propose a novel metadata description language, MML -Metadata Modeling Language, to describe, encapsulate and translate heterogeneous metadata into a canonical format. Moreover, an integration architecture named VUCS@NCTU is also proposed herein. This architecture extracts structured data from digital libraries according to the common structure. On the next step, the extracted data are collected and sent back to user and finally integrated in VUCS server. To verify our idea, we design and implement a virtual union catalog system of university library Webpac systems to fulfill our architecture.

Keywords: Distribute Environment, Digital Libraries, Metadata Modeling Language, Virtual Union Catalog System

二、緣由與目的

數位圖書資訊系統的發展改變了人們對於圖書館的傳統印象。許多原本必須要親自前往圖書館辦理的事情，例如圖書資訊的查詢以及預借等，如今透過數位化的服務，可以輕鬆的在網路上達成。然而數位圖書資訊系統的發展面臨了兩個主要問題：一、數位圖書資訊由不同圖書資訊機構開發而成，因此不同的數位圖書資訊服務常常擁有不同格式的數位資訊。不同的數位資訊並沒有共通的原則因而造成資訊無法互相流通的情形出現。二、不同的數位圖書資訊系統缺乏共通性的整合架構。例如各個圖書館開發的館藏系統，雖然目的相同，卻缺乏共通的整合機制。這些問題肇因於沒有專門為數位圖書資訊系統所設計的架構。一個優秀的數位圖書資訊服務架構，除了可以解讀不同的數位圖書資訊系統的資料之外，還要提供整合的方法，讓類似功能的資訊系統互相整合成為聯合服務的架構。為此，我們提出了一套適用於分散式數位圖書資訊系統整合架構，利用詮釋資料描述語言描述數位圖書資訊的資料，並且透過轉換的機制將不同的數位圖書資訊資料轉換成共通的形式。如此一來，解決了數位圖書資訊格式不同所帶來的問題。

此外，由於全球資訊網 WWW 的發展，幾乎所有的數位圖書資訊系統都採用網頁的方式來呈現資料。館藏系統就是一個最典型的範例。於是我們提出一套階層 ID 的演算法，利用結構化網頁文件具有共通結構的特性，將各自開發的數位圖書資訊系統共同結構的資訊擷取出來，包裝成全式資料描述語言的格式，透過分散式物件傳遞的方式，送至虛擬聯合目錄伺服器整合。此資訊系統整合架構我們稱之為 VUCS@NCTU。本篇論文則是從理論以及實作兩方面的角度共同探腦 VUCS@NCTU 架構的可行性。期望能夠解決上述數位圖書資訊系統資料異質性以及架構共通性的問題。

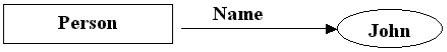
三、結果與討論

詮釋資料描述語言(MML – Metadata Modeling Language)

在本計劃中，我們提出了一套詮釋資料描述語言(MML – Metadata Modeling Language)用來描述數位圖書資訊環境中的詮釋資料。MML 擁有兩樣主要特性：一、豐富的詮釋資料描述能力。二、不同詮釋資料之間的轉換機制。

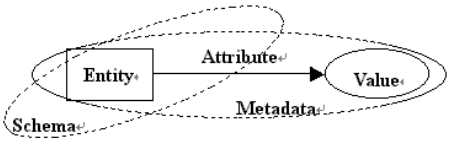
豐富的詮釋資料描述能力

MML 定義了詮釋資料描述的方法。其中包括 MML Data Model 以及 MML Schema 和 Metadata。在 MML Data Model 當中，所有物件的構成都是以 Resource 的型態存在。每個 Resource 是由十個特性所構成[1]。物件的描述是透過三種 Resource – Entity, Attribute 以及 Value。這三者構成 MML 的 Data Model。舉例來講，一個人(Person)的名字(Name)為 John，則在 MML Data Model 當中則描述成如圖一所示。



圖一 MML Data Model

圖二表示將 MML Data Model 描述成詮釋資料所必須具備的兩個部分。MML Schema 代表一個詮釋資料的綱目，用來描述一個 MML 資料所具備的格式以及限制。MML Metadata 則是根據 MML Schema 將資料以 XML 語法 (extensible Markup Language)包裝出來的詮釋資料。MML Schema 以及 MML Metadata 的範例如附錄所示。



圖二 MML Schema 和 Metadata

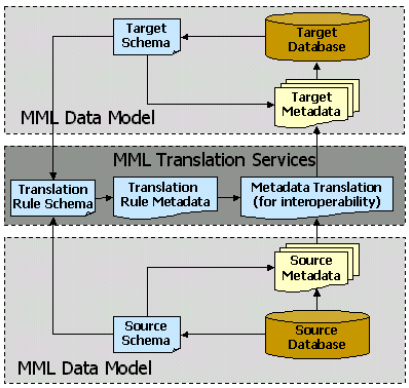
詮釋資料的轉換服務

在 MML 當中，不同的詮釋資料之間是透過 Translation Rule 來做轉換。Translation Rule 是由 MML 所描述而成，其中包含下列三項：來源詮釋資料、目的詮釋資料以及轉換方法。在 MML 的轉換機制當中，我們提供了方便的型態轉換 (如圖三)。MML 轉換機制的範例如附錄。

Basic Operations in MML	
Type	Operation
Integer	Add, Sub, Multiply, Divide
String	String Concatenate
Float	Add, Sub, Multiply, Divide
Boolean	And, Or

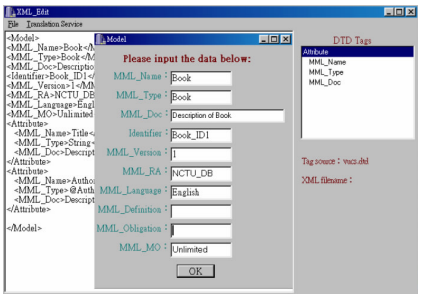
圖三 MML Operations

圖四表示出不同的詮釋資料之間如何達到互通性(Interoperability)的方法。來源詮釋資料(Source Metadata)透過詮釋資料的轉換服務，結合定義好的轉換詮釋資料(Translation Rule Metadata)轉換成目的詮釋資料(Target Metadata)。不同的詮釋資料之間都可以透過詮釋資料交換服務轉換成共通的格式。如此一來便解決了資料異質性的問題。



圖四 詮釋資料的互通性

圖五是我們針對 MML 所開發出來的 MML 編輯器。透過編輯器我們可以編輯 MML Schema，MML Metadata 以及 MML Translation Rule。我們並開發詮釋資料轉換服務的 API，提供給相關程式開發設計師轉換不同的詮釋資料。



圖五 詮釋資料編輯器

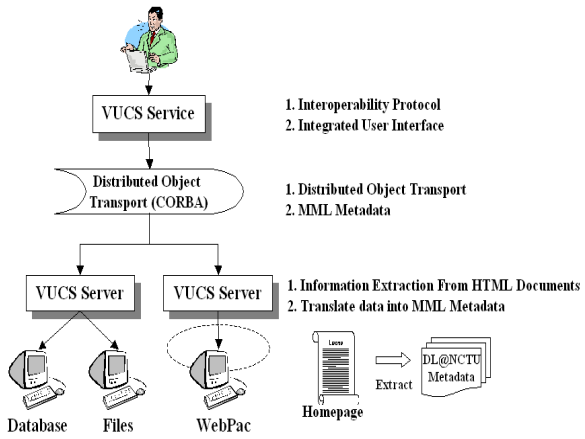
虛擬聯合目錄架構 VUCS@NCTU

虛擬聯合目錄架構主要是用來整合分散式的數位圖書資訊系統。交通大學所發展的虛擬聯合目錄查詢架構(Virtual Union Catalog System Architecture – VUCS@NCTU，透過分散式物件的傳輸方式(CORBA)將由詮釋資料描述語言所描述的詮釋資料傳遞給 VUCS 伺服器(VUCS Server)來查詢所選取的服務。VUCS@NCTU 主要分成三大部分(如圖六)：

VUCS 服務(VUCS Service)

VUCS 服務的功能最主要是提供使用者介面讓使用者從事服務的查詢以及整合回傳後的結果。使用者選擇服務的項目以後，VUCS 服務即將命令以 MML 詮釋資料的方式包裹，傳遞給分散式物件傳輸系統。當分散式物件傳輸系統回傳資料以後，由於回傳的資料符合 MML 格式並且以 XML 當成語法，所以可以套用 W3C (World Wide Web Consortium) 所制定的 XSL(Extensible Stylesheet

Language)，針對回傳的資料作各個欄位排序的變換。



圖六 VUCS@NCTU

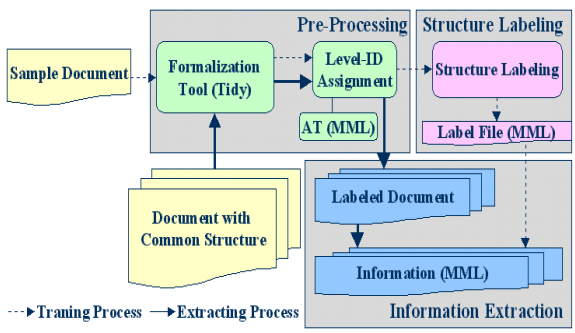
分散式物件傳輸(CORBA)

分散式物件傳輸主要透故 CORBA 來包裹 MML 詮釋資料並且傳送至 VUCS 伺服器。採用分散式物件傳輸可以將物件傳輸透明化，交給 CORBA 負責跨平台的資料傳遞。並且可以透過 CORBA 將分散式物件傳遞的負載量作自動的分配。

VUCS 伺服器(VUCS Server)

一個虛擬聯合目錄架構必須要能從各個不同數位圖書資訊系統中萃取資料，但由於並不是所有的資訊查詢系統都允許直接存取其底層的資料庫，為了讓虛擬聯合目錄系統能夠擷取想要的資料，VUCS 服務提供透過查詢系統所回傳的網頁做資料擷取的動作。資料擷取的工具是由交通大學所發展出來的結構化文件擷取演算法(Extractor@NCTU)，利用網頁之間階層的特性找出我們想要的內容。並將資料以 MML 的語法包裹後回傳。由於回傳的資料皆是 MML 的格式，容易針對回傳的資料作整合。

經過分析發現，網頁具有結構的特性，可稱之為類結構化文件(Semi-Structured Document)。透過階層的分析，我們可以把網頁中具有共同結構的資料擷取出來，我們稱之為 Level-ID 演算法。其目的在於分析文件結構，賦予每個有意義的結構化階層一個 Level-ID 以及相關的語意。透過 Level-ID 擷取演算法，我們可以把相通階層的資料粹取回傳給系統。Level-ID 共分為訓練步驟以及粹取步驟(如圖七)。



圖七 資料粹取步驟

在訓練步驟時，我們從欲萃取資料的文件中，選取一個範例文件（Sample Document)來當作系統的訓練文件，系統會先透過格式化工具(Formalization Tool）將文件做格式化，並參照輔助表格（AT--Auxiliary Table）以分析出文件的結構，接著提供提供一個介面讓使用者標記欲萃取的資料欄位（Structure Labeling)，並利用 MML 語法來描述使用者所標記的資料以儲存成一個結構標記檔（Label File）以供萃取步驟使用。

執行完訓練步驟後，接著讀入和已訓練過的文件具有同樣結構的文件 (Document with Common Structure)，利用格式化工具以及配置 Level ID 以分析文件結構後，參考之前所儲存的結構標記檔以做資料萃取，系統會根據已儲存的結構標記檔作為欲萃取的資料欄位，並利用 Level-ID 演算法來作分析以找出含有相同結構的資料欄位，而最後萃取出來的資料也同樣利用 MML 來加以描述儲存。

Level-ID 演算法

結構化文件最大的特性就是擁有階層式的架構，比如說一本書的章節或段落。階層式的架構可以用樹的型態來表示，每個節點都是代表一個元素，而每個節點所包含的資料就是元素的內容，由圖表四可以更清楚地看出其架構，最上層的節點就是根節點，根節點可能有數個第二層的子節點，而第二層的子節點也同樣可能含有數個第三層的子節點。在此，我們定義兩個性質來幫助在 Level-ID 方法中做文件結構的分析。

定義一、階層性質（Level Property（LP））節點 A 和節點 B 是階層式架構中的兩個節點，階層性質在 A 是 B 的祖先（Ancestor）且文件中 A 的內容（Content）中包含了 B 時成立，此時， A 和 B 具有階層性質的關係。

定義二 平行性質（Parallel Property（PP））節點 A 和節點 B 是階層式架構中的兩個節點，平行性質在 A 和 B 位在階層式架構中的同一層時成立，此時， A 和 B 具有平行性質的關係。

定義好上述兩種性質之後我們就可以分配 Level ID 給每個關鍵元素，Level ID 的格式如下：Level1-ID.Level2-ID.Level3-ID. …

每一層由各自的數字來表示，Level1-ID 所代表的就是第一層的數字，中間的分隔號我們用逗號來表示，其意義就是代表著分層。一個階層式的架構就可以利用這樣的方式來將所有的節點表示成 Level ID 的格式，要完整地將一個結構化文件用 Level ID 來格式化，還必須搭配上一個輔助的表格(Auxiliary Table，如圖八）來記載標籤與標籤之間的階層關係性質，這樣就可利用階層性質與平行性質來正確地配置 Level ID 給每一個元素，輔助表格如圖表五所示，在此，我們以 HTML 文件的標

籤來說明。

輔助表格的用法如下。在分析一結構化文件時，若文件中的一上層元素接下來所出現的元素屬於下層元素時，即代表在此結構化文件存在一個正向的階層性質（Positive Level Property）。相反的，假如下層元素後方出現上層元素時，此時存在一個反向的階層性質（Negative Level Property）。當階層性質出現時，代表結構化文件出現階層結構的現象，此時在 Level ID 的配置過程當中則需要往下擴展一個階層（Down Level）、或者是回到原來所屬的階層(Up Level)。而當前後所出現的元素屬於輔助表格同一列且相同的元素時，即代表平行性質存在於結構化文件。此時將 Level-ID 加一代表屬於同一個階層。

輔助表格		
編號	上層元素	下層元素
1	<HTML>	<Head><Body>
2	<Frameset>	<Frame>
3	<Body>	<Table><Ol><Ul><Dir> <P><Hr>
4	<Table>	<Tr>
5	<Tr>	<Td>
6	<Td>	<Table><Ol><Ul><Dir> <P>
7	<Ol><Ul><Dir>	<Li>
8	<Li>	<Table><Ol><Ul><Dir> <P>
9	<Hr>	<Table><Ol><Ul><Dir> <P>

圖八 輔助表格

前置處理階段（Pre-Processing）

在前置處理階段中，必須先將原來的文件重新格式化並利用 Level-ID 演算法來進行編號，因此前置處理階段主要有兩個部分，分別是格式化工具和配置 Level ID 。

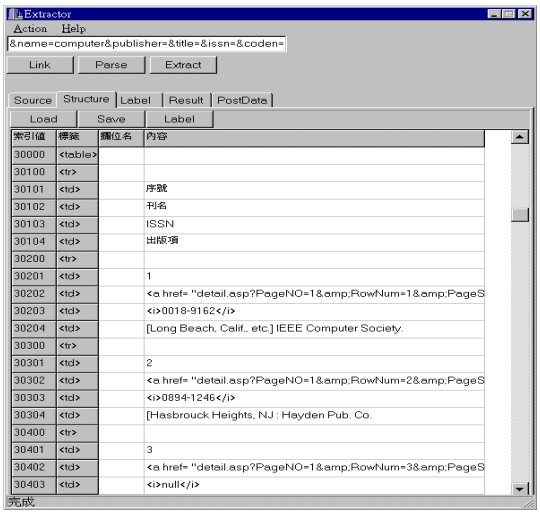
格式化工具（Formalization Tool）

在網路上的文件大都是用標籤語言所描述而成，而撰寫標籤語言最常犯的錯誤就是遺漏了結束標籤或是巢狀標籤的不平衡。這種情形在 HTML 文件中尤其常見，因為 HTML 文件對整個標籤結構並沒有很嚴格地規定，即使網頁撰寫者編寫了不正確的文件，使用者仍然可以從瀏覽器上看到正確的結果。但是對於要分析文件結構而言，這種不正確的標籤文件會導致不正確的結果，因此必須先對讀進來的文件作格式化的動作。在本文中，我們使用 TIDY 這個由 World Wide Web Consortium 所發展出來的網頁格式化工具程式，TIDY 是一個免費的工具程式專門用來找出標籤文件中的錯誤語法，並將之修正成標準的語法。透過 TIDY 的格式化，文件的語法錯誤便可以被消除掉，則我們的系統在分析文件結構時就不會產生錯誤。

配置 Level ID (Level ID Assignment)

經過 TIDY 格式化之後的文件還必須經過配置 Level ID 的動作才可以做分析的動作。我們先對文件的標籤語法作分析，並利用 Level-ID 的演算法配置一個 Level ID 給每個關鍵元素。在我們的系統架構中，一個 Level ID 包含的資訊就相當於此元素在文件裡的階層式結構，這是因為前述的 Level-ID 演算法就是根據整個文件的階層式架構所設計的，只要掌握住某元素在階層式架構中的位置，就能將該元素與結構標記檔做比對分析而萃取到使用者所需要的資料。

圖九就是一個文件在經過格式化與配置 Level-ID 的結果，因為該文件含有重複出現的結構，故其規則性可以很容易地被發現。從圖表七中可以很清楚地看到，文件的序號、刊名、ISSN 以及出版項欄位被分析出來，而其包含的資料也被分析出來並按照欄位順序排列，這些事先的分析讓後續資料萃取的工作更形容易。



圖九 Level-ID

結構標記階段（Structure Labeling）

配置完每個元素的 Level ID 之後，我們設計了一個介面讓使用者標記欲擷取資料欄位的 Level ID，使用者可以點選欲擷取資料的欄位並替每一個選取的欄位命名，系統會記錄下來所有被命名的欄位及其 Level ID，並存成結構標記檔。有了這個結構標記檔，便可以和擁有同樣結構的文件做分析並萃取資料。處理完結構標記之後，就完成了資料萃取的訓練步驟，我們已經擁有了一個可用來當作資料萃取的結構標記檔，接下來就進入資料萃取的萃取步驟。

資料萃取階段（Information Extraction）

在訓練步驟後，我們就可以參照已儲存之結構標記檔來和讀入的文件作比對分析真正把資料從文件中萃取出來。由於已經配置給每個關鍵元素唯一的 Level ID，因此我們可以從結構標記檔與欲萃取文件的 Level ID 特性很容易地萃取出資料。

我們針對各大圖書館的館藏系統設計並實作了一個從含有共同結構之網頁文件作資料萃取的系統，從事虛擬查詢。查詢所回傳的結果如圖十所

示。(參考位置: http://140.113.39.184:9999/)

圖十 館藏虛擬聯合目錄

結果與討論

在本計劃中，我們提出了一套適用於分散式數位圖書資訊系統整合架構，利用詮釋資料描述語言描述數位圖書資訊的資料，並且透過轉換的機制將不同的數位圖書資訊資料轉換成共通的形式。如此一來，解決了數位圖書資訊異質性以及互通性的問題。

透過本計劃，我們可以觀察到兩個問題。第一、數位圖書資訊環境的詮釋資料標準的互通性。有了共通的標準才能使詮釋資料充分交換。近年來有關於詮釋資料的研究有 Dublin Core，Resource Description Framework 等，雖提供標準化描述方式但是相對的也侷限了詮釋資料多變的本質。透過本計劃中提出的 MML 轉換的機制，我們可以兼顧標準以及多樣性的本質。不同的詮釋資料可以用本身的方式來呈現，當需要轉換成標準時，才透過轉換服務來變換。二、數位圖書資訊基礎架構的建立。數位圖書資訊環境可說是一們高深的系統整合學問。除了發展符合使用者需求的服務外，如何將發展完成的系統聯合起來發揮加倍的成效，更需要系統在設計之初就採用合適的數位圖書資訊基礎架構。本計劃提出的數位圖書資訊架構，從資料描述到整個系統的建立以及整合，都完整規劃出一套可行的方案，並且透過館藏系統虛擬聯合目錄來驗證本計劃的可行性。如果能基於本計劃所提兩樣基礎發展數位圖書資訊服務，將能使得數位圖書資訊環境提升到更符合使用者需求的層次。

五、發表文獻

[1] Su-Shang Huang, Hao-Ren Ke, Wei-Pang Yang, “Interoperability of Cooperative Databases with Metadata”, The Fourth World Multiconference on Systemics, Cybernetics and Informatics, July 2000.

[2] Su-Shang Huang, Hao-Ren Ke and Wei-Pang Yang, “Infomration Extraction for Documents with Common Structure”, The Third International Conference of Asian Digital Library, December 2000.

[3] 曾志軒，黃夙賢，柯皓仁，楊維邦，*虛擬聯合目錄系統中擁有共同結構之網頁文件資料萃取*，TANET2000.

[4] 柯皓仁，黃夙賢，楊維邦（2001/03），詮釋資料與數位圖書館系統互通性之探討，大學圖書館 5卷1期（民國90年3月），頁49-78.

[5] 曾志軒，黃夙賢，柯皓仁，楊維邦（2001/01），虛擬聯合目錄系統中擁有共同結構網頁文件之資料萃取,網際網路技術學刊（Journal of Internet Technology），2（1):59-68