

國立交通大學

資訊學院 數位圖書資訊學程

碩士論文

運用 Wiki 建立主題式資源指引網站之研究

A Study of Subject Gateway and Its implementation by Wiki



研究生：陳明燁

指導教授：柯皓仁 博士

中華民國九十六年七月

運用 Wiki 建立主題式資源指引網站之研究

A Study of Subject Gateway and Its implementation by Wiki

研究生：陳明燁

Student：Ming Yeh Chen

指導教授：柯皓仁

Advisor：Hao-Ren Ke

國立交通大學

資訊學院 數位圖書資訊學程



A Study of Subject Gateway and Its implementation by Wiki

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in

Digital Library

July 2007

Hsinchu, Taiwan, Republic of China

中華民國九十六年七月

運用 Wiki 建立主題式資源指引網站之研究

學生：陳明燁

指導教授：柯皓仁 博士

國立交通大學 資訊學院 數位圖書資訊學程碩士班

摘要

主題式資源指引(Subject Gateway)可說是學術圖書館業務的一項特色，圖書館館員粹選與綜整各領域的相關知識，且透過有系統的編纂方式呈現。傳統主題式資源指引的建置與維護需投入大量的人力與心力。有鑑於此，本論文企圖探討以 Web 2.0 的重要工具--維基引擎 (Wiki Engine) —來建置主題式資源指引的可能性。

Web 2.0 是種觀念的改變，所強調的並不在於技術，強調的是『互動與分享』的特質。它主要是以網路為平台，資訊的流向為網站建置者與使用者雙方互動的模式進行、用戶分享共同的創作成果、其成果更是為集體智慧的結晶。本論文利用維基引擎來實際應用在主題式資源指引網站，探討應用維基引擎建立主題式資源指引網站所將面臨的問題，並針對這些問題提出一套可行的解決方案。利用維基引擎的特性，可以讓使用者很簡單地參與內容的建置；而為了維護內容的品質，更是引入了學科專家的機制來一同把關。在研究過程中，研究者實際架設一套開放原始碼的主題式資源指引軟體--Research Guide，比較傳統內容管理系統與 Web 2.0 概念內容管理系統之差異性。本論文所使用的 Wiki 系統平台乃是以 ccTiddly Engine 為基礎，並根據交通大學圖書館的使用需求加以修改與擴充相關功能，以求建置一個通用且適合使用的系統。

關鍵字：主題式資源指引網站、學科資源網、Web 2.0、維基、維基引擎、ccTiddly

A Study of Subject Gateway and Its implementation by Wiki

Student: Ming-Yeh Chen

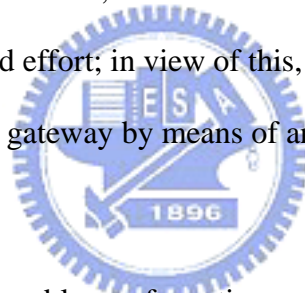
Advisor: Dr Hao-Ren Ke

Degree Program of Computer Science

National Chiao Tung University

ABSTRACT

One distinguished service of an academic library is a subject gateway. Librarians choose essential knowledge of specific subject fields, and systematically present those pieces of knowledge to users. However, the creation and maintenance of a subject gateway incurs a great amount of time and effort; in view of this, this thesis attempts studying the possibility of creating a subject gateway by means of an important tool of Web 2.0 – the Wiki engine.



In order to understand the problems of creating and maintaining a subject gateway in a traditional manner, the researcher creates an exemplary subject gateway using an open-source software, the Research Guide. In addition, the pros and cons of creating a subject gateway using a Wiki engine are also detailed. A Wiki engine allows users to easily participate the contribution of content of a subject gateway. On the other hand, the free contribution of a Wiki engine necessitates a quality-control mechanism.

On the basis of a Wiki engine – ccTiddly, the research takes into account the requirements of the National Chiao Tung University (NCTU) library and develops a Wiki-based subject gateway.

Keywords: Subject Gateway, Path Finder, Web 2.0, Wiki, Wiki Engine, ccTiddly

誌謝

一路上所需要道謝的人真不少，首先感謝的是我的指導教授柯皓仁老師，於撰寫論文時的悉心幫助與指導，不僅提供本論文巨細靡遺的修正，也給予許多寶貴的意見，讓我能一窺學術研究的初貌，受益良多；感謝口試委員謝建成、黃明居、林瑞盛老師們對於論文提供的諸多建議，讓本論文更臻完善；謝謝專班的各位學長姐們及同班同學，當然更謝謝同舟共濟的小組團隊，承嘉上台的 Presentation、和阿華共同架設 Research Guide 系統、婉珍及毓真所負責整理的書面報告、與 Seki 共同 Project、共同修 Wireless and Mobile Network 的雪菁；也謝謝職場上的朋友，特別謝謝瑞娟學姐於公於私的鼓勵與照顧，謝謝一切的點點滴滴，感恩... 感恩。

當然最為感謝的是我的家人，首先感謝父親與母親，他們的鼓勵是我前進的最大動力，沒有他們的支持就沒有今日的我；六年的學涯生活，回到家後總是有熱騰騰的宵夜可以吃(這讓我胖了不少)，無私的陪伴在我左右，打理著許多生活上的細節，也照顧我的兩個調皮蛋，這要謝謝我的老婆；寫作之中可愛的瑋瑋與瑞瑞常常會招受到「颱風尾」的肆孽，可愛的屁屁總是多了些細細的紅線條，希望他們以後若看到老爸所寫的這本論文時，可不要喚起慘痛的回憶，不過說到底小寶貝們也是有貢獻的~讓老爸紓解壓力，也是要稍微誇獎與感謝一下的啦，當然也謝謝弟妹給予的協助。

上台容易緊張的我，總是上台後胡言亂語，鬧了不少笑話，還記得那個搞笑的報告名言嗎？就是那個「Important」的說，English is very important、Java is Very Important and ... ^^。最後應該改成 Friend is very important，畢業後希望能繼續保持連絡，二十年後也許來一場同學會，再來一次台東之旅 ^^，最後祝大家幸福美滿、健康快樂。

目錄

中華民國九十六年七月	I
摘要	II
誌謝	III
目錄	IV
圖目錄	VII
表目錄	IX
一、緒論	1
1.1 研究背景與動機	1
1.2 研究目的與問題	4
1.3 研究流程	4
1.4 論文架構	7
1.5 名詞解釋	8
1.5.1 Subject Gateway	8
1.5.2 Web 2.0	8
1.5.3 Content Management System	8
1.5.4 Wiki	9
1.5.5 Wiki Engine	9
1.5.6 Wiki Syntax	9
1.5.7 Mashups	9
1.5.8 IETF RFC2731	9
1.5.9 CSS	10
1.5.10 OpenID	10
二、文獻回顧	11
2.1 主題式資源指引網站	11
2.2 建置主題式資源指引網站之準則	12
2.3 網路電子資源	16
2.4 傳統主題式資源指引之問題研究	21
2.4.1 主題式資源指引概況	21
2.4.2 主題式資源指引現行問題調查方式	22
2.4.3 主題式資源指引現行問題與改善策略	23
2.5 建置主題式資源指引的軟體	26
2.5.1 ROADS 系統	26
2.5.2 ResearchGuide 簡介	27

2.5.3 ResearchGuide 系統需求與功能.....	27
2.5.4 ResearchGuide 應用實例.....	30
2.5.5 ResearchGuide 測試與評估.....	31
2.6 WEB 2.0 與 WIKI.....	34
2.6.1 Web 2.0.....	34
2.6.2 Wiki.....	39
2.6.3 Wiki Engine	41
2.6.3.1 TiddlyWiki Engine	45
2.6.3.2 ccTiddly.....	46
2.6.3.3 ccTiddly 的系統需求及系統安裝.....	47
2.7 WIKI 在圖書館的應用.....	50
2.8 WIKI 應用於主題式資源指引網站之探討.....	55
三、 研究設計.....	62
3.1 主題式資源指引網站服務現況分析.....	62
3.2 以 WIKI 實作主題式資源指引之問題.....	64
3.3 WIKI ENGINE 挑選	67
3.4 WIKI 實作主題式資源指引網站改善策略.....	69
四、 系統設計.....	76
4.1 系統設計的目標與規劃.....	76
4.2 系統架構.....	77
4.2.1 外部界面(Front End).....	79
4.2.2 中間層(Middleware).....	79
4.2.3 後端層(Back End).....	79
4.3 WIKI 實作主題式資源指引網站改善策略.....	80
4.3.1 管理者觀點.....	80
4.3.2 使用者觀點.....	86
五、 系統實作與驗證.....	87
5.1 系統環境.....	87
5.2 系統功能.....	88
5.3 實作過程.....	90
5.3.1 入口網站.....	91
5.3.2 認證功能.....	92
5.3.3 中文化.....	93
5.3.4 資源的推薦.....	94
5.3.5 詮釋資料與搜尋功能(Metadata / Search Function).....	95
5.3.6 審核功能.....	96
5.3.6 混搭(Mashups).....	97

5.3.7 統計功能.....	100
六、研究結論與建議.....	101
6.1 結論.....	101
6.2 建議.....	103
參考文獻.....	106
附錄一.....	109
附錄二.....	110



圖目錄

圖 1-1 全球網站統計圖.....	1
圖 1-2 國內網站伺服器累計數量.....	2
圖 1-3 研究流程圖.....	6
圖 2-1 IETF RFC2731 著錄 HTML 文件.....	19
圖 2-2 康乃爾大學圖書館以 CORC 來收錄電子資源工作流程圖.....	20
圖 2-3 線上問卷調查表.....	23
圖 2-4 RESEARCHGUIDE 系統架構.....	28
圖 2-5 RESEARCHGUIDE 網頁架構.....	29
圖 2-6 主題指引列表頁.....	32
圖 2-7 單一主題指引.....	32
圖 2-8 學科專家介紹.....	33
圖 2-9 檀香山的 WIKI WIKI 站牌.....	40
圖 2-10 WIKI ENGINE 開發程式語言比例圖.....	43
圖 2-11 TIDDLYWIKI 在 GOOGLE TREND 曲線圖.....	46
圖 2-12 CCTIDDLY V0.5.5 的 DEFAULT.PHP 內的部份程式碼.....	48
圖 2-13 CCTIDDLY 的目錄結構圖.....	48
圖 2-14 使用 WIKI 最常出現之問題長條圖.....	59
圖 2-15 維基軟體處理維基語法之示意圖 [35].....	60
圖 3-1 MEDIAWIKI、TIDDLYWIKI 與 CCTIDDLY 比較表.....	68
圖 3-2 認證模式流程圖.....	70
圖 3-3 審核流程圖.....	71
圖 3-4 CCTIDDLY 資料庫綱要.....	72
圖 3-5 調整後之資料庫綱要.....	72
圖 3-6 SGWS 審核工作流程.....	74
圖 4-1 SG WORKFLOW DIAGRAM.....	77
圖 4-2 SGWS 系統架構.....	78
圖 4-3 程式碼 /CONFIG/DEFAULT.PHP.....	81

圖 4-4 TIDDLYWIKI 風格呈現之結構.....	84
圖 4-5 TIDDLYWIKI 的多風格呈現.....	85
圖 5-1 系統實作環境.....	88
圖 5-2 系統功能圖.....	90
圖 5-3 網站入口檢索頁面.....	92
圖 5-4 WIKI 入口頁面.....	92
圖 5-7 搜尋引擎之改善.....	95
圖 5-8 學科專家審核流程.....	96
圖 5-9 館員審核流程.....	97
圖 5-10 AJAX RSS NEWS	98
圖 5-11 引入 SGWS 系統的 SCRIPT	98
圖 5-12 線上電子字典功能.....	99
圖 5-13 GOOGLE CALENDAR 功能	99
圖 5-14 審核統計功能.....	100
圖 5-15 AWSTATS 報表	100



表目錄

表 2-1 DUBLIN CORE 欄位.....	18
表 2-2 指引評分表.....	23
表 2-3 RESEARCHGUIDE 基本資料.....	28
表 2-4 RESEARCHGUIDE 應用比較.....	30
表 2-5 RESEARCHGUIDE 測試環境.....	31
表 2-6 WEB1.0 與 WEB 2.0 觀念比較表	35
表 2-7 WEB 2.0 網站類型統計.....	36
表 2-8 WIKI 引擎開發程式語言統計表.....	42
表 2-9 排名前十大的 WIKI 引擎.....	44
表 2-10 CCTIDDLY 系統檔案結構及功能.....	49
表 2-11 資料庫表格 TIDDLY_WIKI_ENTRY	49
表 2-12 資料庫表格 TIDDLY_WIKI_ENTRY_VERSION.....	50
表 2-13 WIKI 在圖書館的應用.....	51
表 2-14 免費 WIKI 主機服務者清單.....	54
表 2-15 WIKI 引擎、CMS 與 WEB PAGE 之工作流程比較表.....	57
表 2-16 WIKI 與 BLOG 比較表	57
表 2-17 使用 WIKI 最常出現之問題統計表.....	59
表 3-1 訪談問題及回應.....	63
表 3-2 WIKI 的優缺點分析表.....	66
表 3-3 以 WIKI 實作主題式資源指引的問題.....	66
表 5-1 網站的架構.....	91

一、緒論

高曼(M.Gorman)在其所著「圖書館學五律新詮釋」(The Five Laws of Library Science)中指出,「圖書館為全人類服務」(Library serve humanity)、「使用科技提升服務品質」(Use technology intelligently to enhanced service)、「承先啟後再創新」(Honor the past and create the future)、「知識傳遞多元化」(Respect all forms by which knowledge is communicated)、保障學術自由研究(Protect free access to knowledge) [1]。杜威 (M. Dewey) 也曾說「將適當的資料,在適當的時間,提供給適當的讀者。」長久以來圖書館投入許多努力收錄與維護網際網路上的有用資源,建置成主題式資源指引網站(Subject Gateway)。本研究將實際了解建立主題式資源指引網站所遭遇的困難及探討解決之道,並嘗試導入當下科技—Wiki 軟體協助該業務的運行,期能改善傳統建置模式所面臨的問題,提出適合且能真正運行之系統。

1.1 研究背景與動機

根據英國 Netcraft 公司長期對網站伺服器數量的調查發現(如圖 1-1 所示),在該公司 1995 年 8 月發布首份調查報告時,僅有 18,957 個網站;2004 年時,其總數到達五千萬左右;到了 2006 年 10 月,全球網站總數已突破一億大關了[2]。

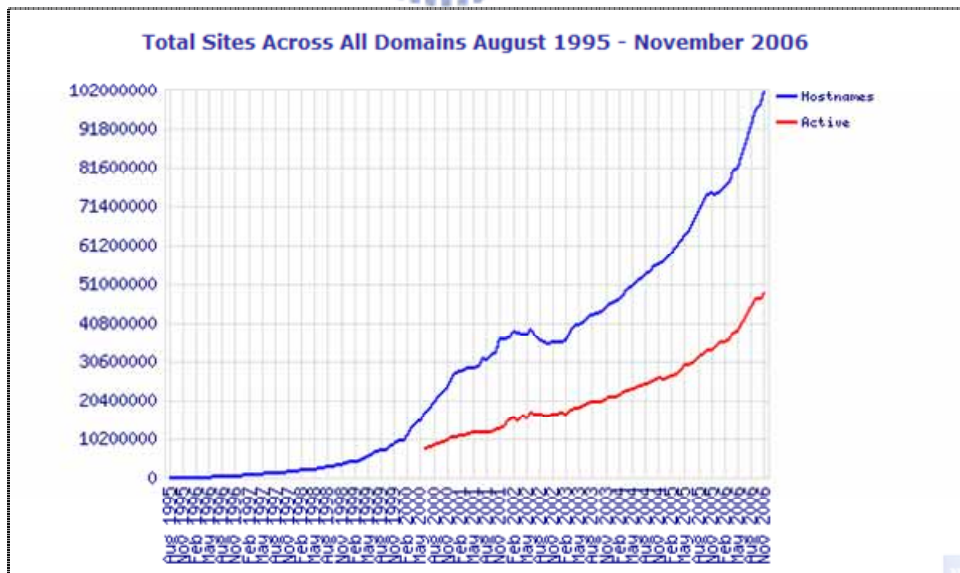


圖 1-1 全球網站統計圖

(資料來源：http://news.netcraft.com/archives/web_server_survey.html 上網日期：

2006/11/27)

而國內網際網路的現況，可自台灣網路資訊中心獲得相關資訊。根據該中心所公布的 2007 年「台灣寬頻網路使用調查」報告(<http://www.twNIC.net.tw/>)，截至 2007 年 1 月 15 日為止，台灣地區上網人口約為 1,523 萬人，整體人口(0-100 歲)上網率達 66.91%，12 歲以上人口的上網率則為 67.77%；其中，寬頻網路使用人數約 1,240 萬人，約佔總人口數 63.36%。而個人上網的目的以瀏覽網頁(67.37%)與收發電子郵件(40.34%)為多數。由該報告中尚得知國內登記有案之網站伺服器累計數量已達 85,217 台(如圖 1-2)，教育學術網域內的網站伺服器主機則計有 3,552 台。

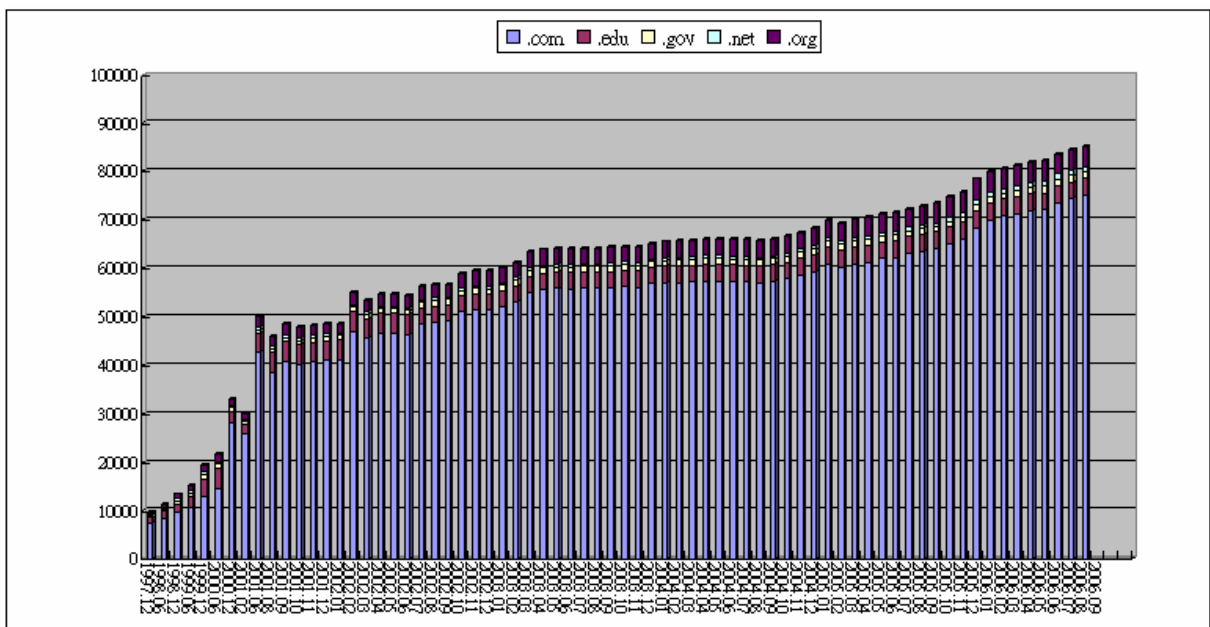


圖 1-2 國內網站伺服器累計數量

(資料來源：http://www.twNIC.net.tw/dn/dn_g_01.htm#3 上網日期：2007/03/20)

網際網路上的資訊數量龐大，其中所涵蓋之內容更是包羅萬象。要在網際網路上有效率地獲得有用的資訊，並不是一件容易的事，僅是搜尋資料可能就需花費很長的時間，更不用說要去理解、消化這些資料。加以網際網路上充滿著似是而非的資料，使用者亦難以判別其正確性。

通常解決網際網路上尋找資料的方式為借助於『搜尋引擎(Searching Engine)』或『專人整理的目錄』。前者的功用在於提供強而有力的搜尋機制，以迅速找到所要的資料，

但這些資料，也許有用，也許會誤導方向，通常所找出的資訊量亦十分可觀。而後者則能讓使用者利用專人整理好的目錄，迅速獲得有用資訊，主題式資源指引網站即屬於後者。

不可諱言地，圖書館所整理的各式主題資源，可讓使用者透過主題式資源指引網站，迅速地指引到優質網站，獲得正確的資訊，能更有效率地正確學習各類學科知識。但以現行建置與維護主題式資源指引網站的作法，主要仍面臨了網路資源異動性、資源連結網址的變動、資源描述的更正、是否符合讀者期待...等問題，最大的瓶頸是在於後續維護作業。圖書館能投入的人力與物力有限，要如何整理這無限的網路資源是一大挑戰。綜上所述，本研究擬探討導入 Wiki 軟體協助圖書館建置主題式資源指引網站之可能性，企圖提出一個以合作模式運行之系統，透過眾人之力進行資料徵集以有效地發掘具價值之網路資源；並藉由圖書館館員或學科專家共同把關，提供高品質之主題式資源指引。最後期能設計一套適合且能真正運行之系統。



1.2 研究目的與問題

圖書館投注了大量的人力與物力建置與維護主題式資源指引網站，最主要的目的在於協助讀者有效且快速地發現網際網路上高品質的資源。本研究的目的是在於應用 Web 2.0 之理念與技術，導入 Wiki 軟體協助該業務運作，為圖書館的主題式資源指引服務導入新的運作模式，將資料的徵集蛻變為相互交流的方式，並設計實際可供運作之系統支援該項業務的進行。基於前述的研究目的，本研究將深入探討以下問題：

1. 現行主題式資源指引網站有哪些窒礙難行的困境？
2. 網路資源的徵集是否有相關標準與策略可供參考？
3. 網路資源目前所廣泛採用的著錄格式為何，要如何導入於 Wiki 軟體內？
4. 以 Wiki 軟體應用於主題式資源指引網站所將面臨那些問題？所需具備的要件為何？



1.3 研究流程

圖書館在維護主題式資源指引網站所面臨的最大困境在於資料的徵集與後續的維護，要如何協助圖書館能以精簡的人力有效地投入該業務，是本研究所要達到的主要目的。

本研究的進行步驟如圖 1-3，主要探討導入 Wiki 軟體建置主題式資源指引網站之可能性。研究流程分成三階段進行，第一階段進行現況分析，實際了解圖書館建置學科資源指引網站的作法，本階段以清華大學各系所學科指引網頁 (<http://www.lib.nthu.edu.tw/guide/>) 為對象，從人力、相關建置技術、工作流程...等角度切入，實際了解現行建置主題式資源指引網站所面臨的挑戰。第二階段則是文獻回顧，主要從主題式資源指引網站、網路資源及 Wiki 三個面向切入探討，首先了解主題式資源指引網站的現行建置方式與相關技術；其次觀察與整理網路資源相關的專案如 CORC(Cooperative Online Resource Cataloging)，發掘可資借鏡之處；最後了解 Wiki 相關應用與技術。透過前二階段可了解現行主題式資源指引網站在建置上的優缺點，以及

將 Wiki 應用於主題式資源指引網站時的基本原則，從而於第三階段實際發展一套將 Wiki 融入主題式資源指引網站建置的系統。

本研究主要依循著 Web 2.0 的精神、理念、技術，並參照康乃爾大學參與 CORC 收錄網路資源的工作流程，嘗試改變傳統建置主題式資源指引網站的工作流程，朝著與讀者/使用者共同參與、互動的模式運作，可讓使用者參與內容的建置與維護。內容的徵集、整理、發掘不再集中在圖書館單方面，而是可分散於各個使用者。使用者有權可以參與維護及建置。在此一模式下所產出的內容，相信更符合使用者的需求，也更符合使用者之心智模型(Mental model)。



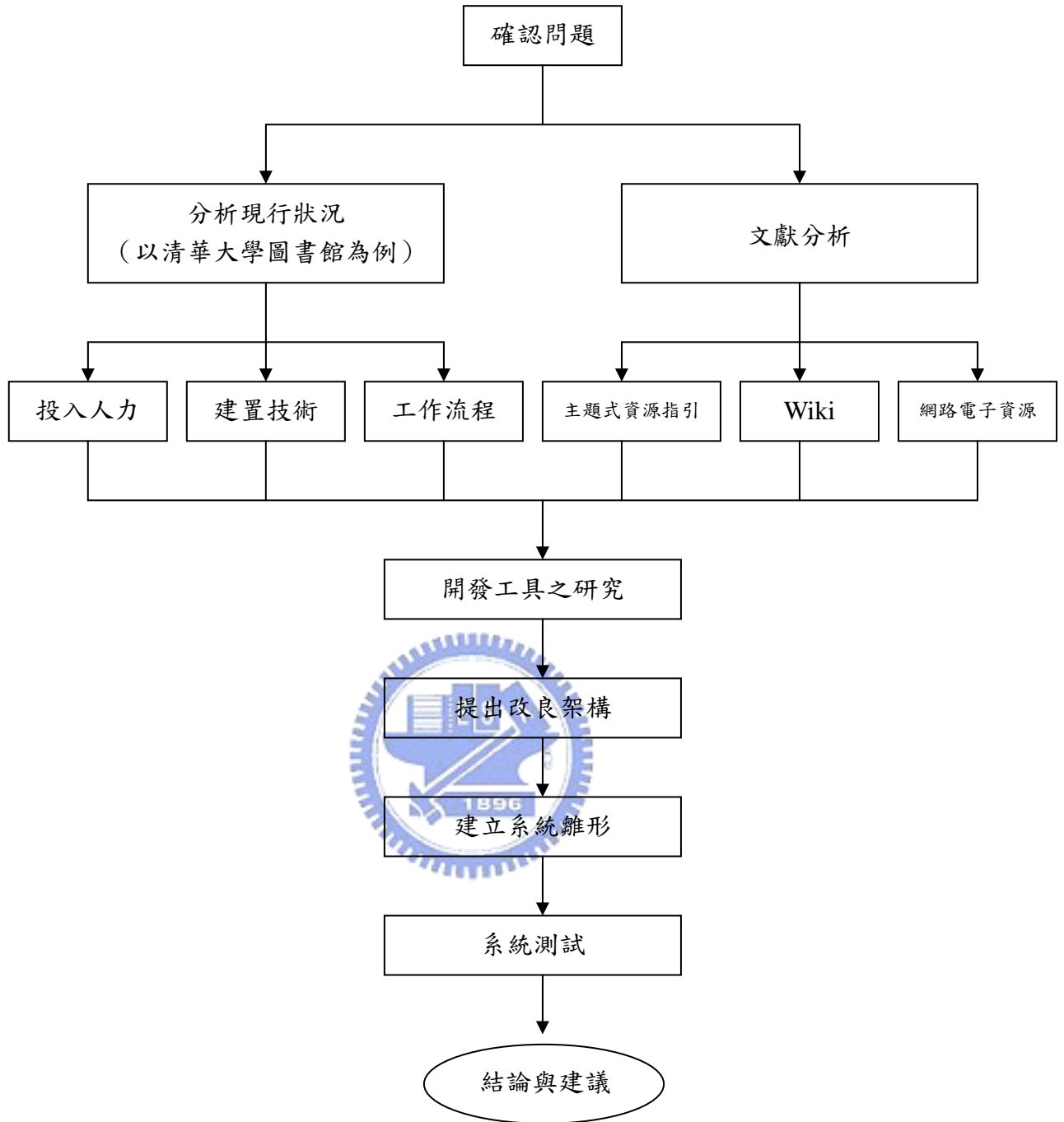


圖 1-3 研究流程圖

1.4 論文架構

本研究共分為六章，各章重點簡述如下：

(一) 緒論

本章先說明論文之研究背景與動機、研究目的、研究流程、論文架構與名詞解釋。

(二) 相關文獻回顧

文獻回顧主要從主題式資源指引網站、網路電子資源及 Wiki 三個面向切入探討：

1. 探討主題式資源指引網站的建置方式及現行作業模式所面臨的困境。
2. 觀察目前圖書館如何整理及著錄網路資源，並參考康乃爾大學以 CORC 系統整理網路資源的工作流程以作為系統開發之借鏡。
3. 研究 Wiki 之相關定義與 Wiki 在圖書館的相關應用。

(三) 研究設計

本章在於說明研究方法。實際了解目前清華大學圖書館所建置之學科資源指引的問題後，嘗試使用 Wiki 軟體輔助主題式資源指引網站的建置，導入 Web 2.0 之觀念，結合大眾之力一同參與，共同編寫的方式，改善現行作業方式。系統平台主要使用 Wiki 引擎為基礎，試以改良，使之更符合圖書館的需求。藉由此一系統平台以求降低館員的工作負擔，並以降低所投入的人力、後續維護、設備...等相關的工作成本，達到服務的最高品質。在此提出與傳統模式之不同的建置方式及運用 Wiki Engine 實作主題式資源指引所會面臨問題之討論。

(四) 系統設計

此章主要在於討論以 Wiki 引擎實作主題式資源指引之解決方案。在此章提出本研究之系統架構，設計出的系統可與圖書館系統的讀者認證模組結合，且將原 Wiki 引擎加入詮釋資料(Metadata)機制，以求改善 Wiki 引擎全文搜尋功能。為確保所收錄資源之品質，更增加了審核功能機制讓學科專家及館員能一同參與品質的控管。

(五) 系統實作與驗證

介紹系統實作的開發環境，展示以 Wiki 引擎來實作主題式資源指引網站的結果，以及對所提出之問題做初步的結果驗證。

(六) 研究結論與建議

闡述本研究之結論，以及給未來研究者之建議。

1.5 名詞解釋

1.5.1 Subject Gateway

主題式資源指引網站(Subject Gateway)，是指經由專家介入，有明確的收錄主題，並可提供使用者獲取經整理、推薦的高品質之網路資源指引，網站內提供詳細的資源描述性詮釋資料以作為搜尋與瀏覽的基礎，並透過網路直接引導至該資源所在處的服務方式謂之[3]。



1.5.2 Web 2.0

Web 2.0 是從 O'Reilly 與 MediaLive 所共同舉辦之國際研討會議中引發討論，改變了傳統的 Web 溝通方式，提供一個嶄新的傳播媒介，Web 上的資訊傳播從傳統的單向進行，演變成可雙向溝通的模式，換言之，Web 2.0 的服務方式是以 Web 為基礎平台，網站內的資訊是開放的，使用者可以很方便、簡易地參與互動。目前 Web 2.0 最受歡迎的工具軟體，就是 Blog 與 Wiki。

1.5.3 Content Management System

內容管理系統(Content Management System)為應用於內容管理的軟體系統，可處理電腦檔案、影像、多媒體，影音檔、電子文件與網頁內容。若是針對網站內容管理的軟體亦可稱為 Web Content Management System，使用者只要透過管理介面即可很方便簡單地完成內容的撰寫、修改、儲存、控制(Controlling)、版本控制(Versioning)、出版產業特定的文件(Publishing industry-specific documentation)。

1.5.4 Wiki

Wiki 一詞，從不同的面向有不同的解釋，若由系統面來觀看，Wiki 是一套可以簡易創造、改變 HTML 網頁的系統。Wiki 允許使用者藉由瀏覽器改變線上所看到的頁面內容，這讓 Wiki 的使用者，可以藉由 Wiki 平台十分簡單且容易的集體合作在同一主題及這些超文件上。此外，Wiki 可記錄所有的更動歷程，並可為每個條目定義其分類(標籤)。

1.5.5 Wiki Engine

Wiki Engine，或稱為 Wiki Software，是指用來架設 Wiki 系統的軟體。廣義來說，即是一種提供網路共筆，供網友自行編輯，並最終集成完整的資料庫；狹義來說，即是能達成維基百科樣式的軟體。

1.5.6 Wiki Syntax

Wiki Syntax 也稱 Wikitext 或 Wiki markup，它主要是用來提供替換 HTML 語法的一種簡單的標記語言(Markup Language)，此語法專用於 Wiki 網站內，例如 Wikipedai 網站即是採用該語法撰寫。



1.5.7 Mashups

混搭(Mashups)也稱 Web Application Hybrid，指網站或網站應用程式混合著其它來源的內容，以提供更為豐富的資訊。通常是藉由公開的 API、Web Feeds(e.g. RSS or Atom)、Javasctipt 來取得內容來源。Programmableweb 網站 (<http://programmableweb.com/mashuplist>) 提供了數千種已公開的公開的混搭程式。

1.5.8 IETF RFC2731

Internet 互聯網工程工作小組 (Internet Engineering Task Force, IETF) 所提出的 RFC 2731 規範定義了採用 Dublin Core 著錄 HTML 文件的方法(Encoding Dublin Core

Metadata in HTML)，撰寫方式可參考 <http://www.ietf.org/rfc/rfc2731>。

1.5.9 CSS

CSS 為 Cascading Style Sheets 的縮寫，是 W3C 所提出的標準，是一種用來定義 HTML 外觀(如字體、間距和顏色)的語言，目前最新版本為 CSS2.1。

1.5.10 OpenID

OpenID 由 Brad Fitzpatrick 於 LiveJournal (Open Source Server Software) 提出，目前版本為 1.1 版，AOL 與 Yahoo 等機構均已支援該作法。它是一種分散式身份認證系統(Decentralized Single Sign-on System)，使用者僅需在支援 OpenID 伺服器內註冊後，就能以該帳號(OpenID Identity)使用在所有支援 OpenID 的網站，取代原有現行註冊與認證方式，該工作流程可參考 [OpenID 1.1 Protocol Flow \(http://openid.net/pres/protocolflow-1.1.png\)](http://openid.net/pres/protocolflow-1.1.png)[4]。



二、文獻回顧

本章主要說明主題式資源指引網站之相關定義、電子資源之徵集方式與 Wiki 相關應用，回顧文獻中歷年來圖書館在實作主題式指引所面臨的問題及解決之道，並參考清華大學圖書館各系所學科指引，了解實務上所面臨之問題。此外，本研究亦架設 ResearchGuide 分析其優缺點，做為本研究之借鏡及發展基礎。

2.1 主題式資源指引網站

許多學術圖書館自 1970 年代起就開始建置資源示意圖 (Pathfinder)。資源示意圖主要是針對特定主題列舉館藏中相關的資源，以供讀者按圖索驥來找尋資料，是圖書館介紹館藏、服務使用者的利器，其優點包含：(1) 將圖書館館藏各類型資料都包含其中，並介紹其查找方式；(2) 提供最切合使用者需求的資源，節省讀者時間，並避免圖書館資源的浪費；(3) 指導利用圖書館，幫助使用者瞭解圖書館資源並學習如何利用圖書館 [5]。而主題式資源指引更是由這個基礎延伸而出的一種服務方式 [6]，主題式資源指引是學術圖書館業務的一項特色，由圖書館館員針對特定的主題範圍有系統地彙整與編纂相關知識 [7]。因應網際網路的普及、時代的變遷，主題式資源指引逐漸轉型為主題式資源指引網站 (Subject Gateway，以下簡稱 SG)。

主題式資源指引網站的定義及名稱，目前因各建置單位所著重的層面不同而會有不同的名稱，包含品質控管型主題式資源指引網站 (Quality-controlled Subject Gateway)、澳洲主題式資源指引網站論壇 (The Australian Subject Gateways Forum)、網路主題資源指引 (Internet Subject Gateway)、資訊入口網站 (Information Portal)、知識入口網站 (Knowledge Portal)、主題式資訊指引 (Subject-based Information Gateway) ... 等，名詞雖不同，但其目標與建置目的均是為了協助使用者尋找高品質的學術性主題資源而建立，Dempsey 等學者則更進一步指出，其實不需拘泥於主題式資源指引網站的精確定義，只要瞭解箇中精神即可 [8]。

歐洲研究和教育資訊發展計畫(Development of a European Service for Information on Research and Education, DESIRE Project)將主題資源指引網站定義為一種網際網路上的線上服務，提供對網路資源做檢索及分類瀏覽，它通常的典型是專注在學術上特定主題相關範圍的收錄[9]。瑞典學者 Koch 對主題式資源指引網站其定義為是一種網路上的服務，支援有系統地發掘大量的網路資源，該類型網站通常是以人工選擇資源，以確保收錄資源的品質，同時也會利用標準的詮釋資料格式(Metadata)來描述資源，網站內提供著豐富、大量具有品質性的優質資源指引，這些資訊定期進行檢查及更新，以確保其品質性，最主要的目標在於提供具控制詞彙的索引(Index)資源，建立主題資源索引藉以提高檢索率，網站內並提供完善的分類架構，藉以協助使用者進行進階搜尋與瀏覽[10]。簡而言之，主題式資源指引網站為一種以支援系統性資源發掘為主的網路服務，其所關注的是收集網路上分散的資源，提供主題分類架構，讓使用者藉以瀏覽與獲取資源。除了提供各種網路資源(包括文件、物件、網站、或服務的連結)，詳盡的資源描述亦為其服務基礎[3]。

國內學者卜小蝶教授綜合各學者之評論，進而歸納整理出以下定義：主題式資源指引可定義為對特定主題或類型之網路資源予以收集、整理，以協助使用者獲取高品質網路資源的指引服務[8]，主題式資源指引網站應包括作為取用網路資源的便捷管道、需要專家介入(包括主題及資訊專家)、清晰的主題範疇(以特定主題為主)、明確的資源選擇標準、一致的資源描述格式、高品質的資源描述內容、清楚的資源建立與維護說明、對資源加以分類與提供主題索引、及提供分類架構協助資源搜尋與瀏覽等特性。簡言之，該項服務為經由專家介入，有明確的收錄主題，並可提供使用者獲取經整理、推薦的高品質之網路資源指引，網站內提供詳細的資源描述性詮釋資料以作為搜尋與瀏覽的基礎，並透過網路直接引導至該資源所在處的服務。

2.2 建置主題式資源指引網站之準則

1990 年代開始，圖書館為了協助使用者尋找專業性資源及過濾大量不相關的網路資源，乃採用兩種不同型態的服務模式：一種是把網路資源視為另一類的館藏資源類型，依本身既有的館藏發展政策予以選擇、收集與整理，甚而建立網路資源合作編目機制；另一種方式則是針對網路資源，由圖書館本身或與相關學術機構尋求合作，另外建

置所謂的主題式資源指引網站[3]。

無論採用何種服務型態，圖書館在整理這些網路電子資源時，是否有一定的標準或規範可供遵循或參考？回顧相關文獻可發覺，各館在引入電子資源時，多依館內既定政策予以收錄相關資源，充實自我典藏。以交通大學圖書館之電子圖書館實施經驗談為例[11]，該館在收錄電子資源時會先做引進前的評估，引進主要考慮因素是依據系所的推薦和圖書館的經費來決定，由電子資源館藏發展綱要可知，主要的評估標準仍是針對資料庫、電子期刊等資源；網路電子資源整理的依據，大多仍列出概念性的條款供館員參考。大致而言，傳統圖書館所發展出的資源評估標準仍適用於電子資源，ARL 研究報告亦指出 54% 的圖書館選擇網際網路上的資源仍採用紙本資料的標準，43% 則採用不同的標準，Nisonger 則認為大部份傳統紙本資料的選擇標準仍是適用於網際網路資源的選擇，但也必須針對其特性發展出獨特的篩選方式[12]，專家學者於 1997 年起陸續提出許多評鑑網路電子資源篩選原則可供參考，當中最具特色與完整的是美國喬治亞大學教育科學系(Gene L. Wilkinson)所完成的篩選標準(The Georgia Criteria)，相關標準也可採用紐西蘭維多利亞 G. Smith A Toolbox of Criteria、J. Rettig and C. LaGuardia 所提出的 Retting/LaGuardia Canon for Reviewing Electronic Resources 或 SCOUG Criteria... 等篩選標準，事實上圖書館要採用何種標準篩選網路電子資源，是沒有任何強制性的，各個標準的差異在於評鑑項目的不同而有所區分，而目前一般所公認所必須注意的網路資源評鑑項目則以權威性(Authority)、正確性(Accuracy)、時效性(Currency)、可及性(Accessibility)、可用性(Usability)、範圍(Scope)、適用對象(Audience level)、獨特性(Uniqueness)、組織/(Organization/Presentation)... 等最為重要[13]。

根據 2.1 節的說明，主題式資源指引網站是一種以網路為本，並以收集高品質、經評鑑，且能支援特定學科主題研究資源的管道。而欲從浩瀚的網路世界中挑選適當的資源，又有那些準則可供遵循呢？在此以 DESIRE 所產出的專案報告來做探討。這份報告主要分成三部份[14]，第一部份主要在於介紹主題式資源指引網站的定義、在建置前的初步工作、系統需求、維護的基本需求... 等；第二部份則是在闡述收錄網路資源時的準則，包含品質的選擇、如何發現資源、詮釋資料的格式、主題索引及分類、典藏的管理、資訊提供者的工作流程、如何行銷、使用者介面等議題的討論；第三部份則在於討論與科技相關的議題，如軟硬體系統需求、如何實作、無障礙等議題。由該報告中，可以了解收錄網路資源的若干建議準則，如資源的發掘、品質的篩選、詮釋資料的格式、如何

建立分類目錄、索引及分類的建立、如何做好典藏管理等。建置主題式資源指引網站的第一步，就是建立本身的發展政策及選擇政策。成立發展政策的好處在於(1)可讓使用者明確的了解網站所成立的目的與宗旨，以及該網站所收錄地資料種類與範圍。(2)由於網站內所收錄的資源是經過品質控制且主題明確，使用者可更方便且有效率的獲得指引。(3)明確的發展政策可讓工作同仁確保所徵集資料前後的一致性。(4)明確的政策也易於使用於提供新人的教育訓練，這也確保由不同的部門來收錄這些資源時能有一準則可以遵循。而選擇政策主要是要用來確保收錄典藏的品質，與提供收錄相關資源的參考標準，而不是依工作人員喜好所決定，通常圖書館也會設立一些建議要點供參考。以下茲列舉 DESIRE 所提出的五點資源發掘策略[14]：

1. 避免重複的工作(Avoid duplicated effort)：重複的徵集相同資源是浪費人力資源，尤其是不同計劃內的主題資源網站與不同的團隊最容易發生這種狀況。單位在建置主題式資源指引網站前，可事先調查相關領域內是否已有相關主題網站，若是發現相關性質主題已建立完成，則必須思考是否值得花費時間與精力來從事相同主題的收錄，或許可考慮與這些已建置相關主題單位進行合作，考慮採以跨搜尋的方式來提供服務，彼此互相分享所收整理的詮釋資料。
2. 尋找正確的人來做正確的事(Find the right people for the job): 通常建置主題式資源指引網站的工作人員，通常有資訊技術人員、志願者、學科專家以及圖書館員等人員的協助。資訊技術人員主要是負責伺服器及網站的建置作業，而網站內的內容則由志願者、學科專家以及圖書館員負責。內容的建置交由志願者建置是最不花費成本的作法，但通常問題也最多；學科專家(Subject specialists) 則能提供最佳的資源指引資料，但是通常專家們不一定有空可以協助網站內容的建置，當然採用此模式所建置的成本也是最高的。由圖書館館員進行資源發掘，則需要額外的教育訓練，主要是因為學科專業背景的不同，當然成本也不低廉，且對新手而言是不容易的。
3. 提供有關資源發掘的教育訓練(Provide training in resource discovery)：網際網路持續地成長與變動，所以必須提供適當的訓練及技巧，協助有效率地發掘資源，

實作方式可以列出目前現有的資考資源清單、良好的範例說明以供參考或利用集體研討的方式來協助。

4. 建置可支援資料發掘的系統(Setup support systems for resource discovery staff)：
另外建置可支援資料發掘的系統，例如將目前網頁資源清單的搜尋策略列出一份清單供工作同仁，可提供線上即時的參考與使用；將參與發掘資源者的郵件清單列出，可讓參與者可藉由電子郵件彼此分享資訊；設立定期的會議，定期的討論使得彼此能分享相關成功經驗。
5. 建置可激勵使用社群建議資源的系統(Setup systems to encourage your user community to suggest resources.)：明確的收錄主題範圍，可讓網站的使用者主動提供回報與協助相關資源的發掘管道，通常是藉由宣傳再透過網站 Web Form 機制達成。

綜合而言明確的定義網站的發展政策與選擇政策是有助於圖書館與網站的使用者，它可讓資料徵集的工作流程更為順暢，進而達成有效率的發掘資源，同時也避免了各個工作團隊產生收錄標準不一致的狀況，當然明確的定義則更能讓使用者清楚圖書館在從事主題式資源指引網站的運作，當然能讓他們明白網站內所能提供的主題資源，而能更有效率獲得幫助與搜尋。

2.3 網路電子資源

本節主要在於討論網路電子資源定義與網際網路所產出的文件是否具參考與價值性，並觀看目前主要整理網際網路相關編目標準，參酌康乃爾大學圖書館收錄與整理電子資源之工作流程，藉以提供本研究之借鏡參考。

網際網路所產出的文件是否具參考與價值性呢？是否值得圖書館組織來整理呢？事實上，網際網路影響著目前的學術環境(Scholarly communities)，改變原有資料蒐集的方式，也提供展新的「溝通」媒體模式，所產出的資訊量更是難以統計，R. Chepesiuk 更指出當網際網路蓬勃發展後，圖書館組織資料的標準、服務與工具必須予以立即發展與修正，才能在這蓬勃發展的網路世界中協助讀者查詢所需的資源[15]。OCLC (Online Computer Library Center)亦指出[16]：

1. 許多有價值的資訊可自網際網路上取得。
2. 這些資訊必須被組織，以利檢索。
3. 以現有圖書館技術與程序編製書目記錄，透過線上目錄的檢索，是取用資源最有效率的方法。



觀看目前於網際網路上的編目整理專案，可發現目前則以主題式資源指引網站 (Subject Gateway) 與 CORC(Cooperative Online Resource Catalog)最為成功。主題式資源指引網站主要是由歐洲發起並進行推廣且發揚光大，CORC 則是由美洲 OCLC 所主導的一項網路資源合作編目計劃，CORC 採用最新技術所開發出來的 Web-based 雛型系統，主要目的是希望透過圖書館間的共同合作，並發展自動化工具藉以輔助網際網路資源的創造、選擇、組織及維護網頁資源，期使網路上龐大的電子資源能獲得整理，協助讀者可更有效率的取得與利用，CORC 於 1998 年 11 月開始開發其原型 (Prototype)，並於該年 1 月份向世界各地徵求志願參與的圖書館，從此 CORC 就成為每天數百人使用的一套線上電子資源編目系統[17, 18]。主題式資源指引網站與 CORC 兩者間最大的差異性在於「分散」和「集中」。主題式資源指引網站採用主題為出發點，實現各個分散不同主題的收錄後，最後在以此為基礎下，推出綜合性的入口，可說是由分散到集中的一種作法；CORC 作法則相反，CORC 在開始收錄時就採用統一的作

業平台，各個圖書館透過單一網頁入口，僅有 OCLC 編目權限者，才能參與主題專家們的共同進行合作編目，該流程與主題式資源指引網站相反，可說是由合而分。但兩者就目的上而言，其共同理念與目標均為相同，但於作法上 CORC 則更適合一主導單位統一來要求進行，如國家圖書館。而主題資源指引則更適於個別機構，自行獨立作業，最後在將各分散各處的相關資源串連起來[19]。

而整理網際網路電子資源的著錄格式多採用都柏林詮釋資料核心集(The Dublin Core Metadata Element Set，簡稱 DC)予以詮釋。以 CORC 系統為例，主要的目標是針對網際網路資源加以編目，藉由世界各地的館員共同為網路資源編目，並共享編目成果，而採用的編目格式就可採用 DC 詮釋[17]。而什麼是 DC 呢？它的用途為何？DC 於 OCLC 與 NCSA 於 1995 年在 Dublin Core 研討會中所提出，主要的目的是針對網際網路的快速發展與資源整理的需求而產生，DC 主要的功用在於提供一個簡單易懂的資源描述集，可提昇資源在跨領域、跨主題的可見度，且建置成本遠較於傳統圖書館所採用的 MARC 格式低廉，故目前已被圖書館、博物館等單位廣泛使用。DC 可針對網際網路資源做一般性的描述，以求跨領域使用者的了解，也可再進一步深入的描述，提供語意更為豐富的描述服務。網路資源的使用者藉由 DC 所提供的檢索詞彙查詢網路資源，可獲得基本的相關資訊及指引。目前採用十五個描述方式詮釋資料，各項功用描述如表 2-1。欄位項目可分為名稱、創作者、主題、資料相關敘述、出版者、其他貢獻者、日期、資料型態、格式、來源、語文、資料內容所包括的時間與空間範圍與智慧財產權，採用 DC 描述時並無必備欄位的要求，各個欄位皆可彈性擴充。

表 2-1 Dublin Core 欄位

欄位名稱	標籤	說明
名稱	Title	資源名稱，資料創作者或出版者所附予資源的名稱。
作者	Creator	資源的作者(Author)或創作者(Creator)，資源內容的負責個人或單位，如文件的撰寫者、相片的攝影者。
主題	Subject	主題或與內容相關的關鍵字，以表達資源內容主題，可透過關鍵詞句表達或是利用現有標題表或分類表。
相關敘述	Description	與資源內容有關的文字敘述，例如文獻摘要或者是視覺資料的內容。
出版者	Publisher	出版現行資料的單位。
其他貢獻者	Contributor	其他任何對資料內容有貢獻的個人或機構，但對資料的貢獻不若作者顯著。
日期	Date	資料以目前型式存在的日期時間。
資料型態	Type	資訊類型，如網頁、小說、詩詞等。
資料格式	Format	資料格式，有利於應用軟體的判斷及資料的使用。
識別號	Identifier	足以辨識資源的數字或字串，如 ISBN。
資料來源	Source	足以辨識資源原始資料的數字或字串。
語文	Language	資源內容所使用的語文。
關連資料	Relation	其他關連資源資訊。
內容範圍	Coverage	資源內容所涉及的時間或地理區。
使用權限	Rights	連結到資料使用說明資訊。

參考來源：[20]

若從資料庫的角度來檢視 DC，它包含可協助或限制檢索的詮釋資料、作為分享與互通之用的詮釋方式、表達資料的特性可作為索引資料的詮釋資料，換句話說 DC 可擴充協助檢索面向，讓資料可以藉由各種不同角度進行檢索，例如採用作者、主題、建立的時間... 等方式尋找，可讓資料的搜尋更為精準，網際網路資源的編目格式採用 DC 進行整理，而網頁也採用 IETF RFC 2731 規範(Dublin Core Metadata) 來著錄超文件標示語言(HTML)(如圖 2-1)，藉以提供更豐富的語意描述，方便搜尋引擎能更簡易的建立索引也提供編目人員更多的語意參考。

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
```

```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<title>交大學科資源網</title>
<link rel="schema.DC" href="http://purl.org/dc/elements/1.1/" />
<meta name="DC.title" content="交大學科資源網" />
<meta name="DC.description" content="「交大學科資源網,SG@NCTU」，此學科資源網徵集並篩選國內外優質學術性網路資源，期使支援各學科領域教學與研究。" />
<meta name="DC.date" content="2007-07-04" />
<meta name="DC.format" content="text/html" />
<meta name="DC.language" content="中文" />
<meta name="DC.creator" content="陳明燁" />
<meta name="DC.publisher" content="國立交通大學圖書館" />
```

圖 2-1 IETF RFC2731 著錄 HTML 文件

國際網路電子資源整理工作流程，可參考康乃爾大學採用 CORC 系統來整理國際網路的方法，該大學主要分成幾個步驟進行整理，並可依其建議繪製出電子資源的收錄過程(圖 2-2)，該工作流程首先由資料挑選者(Selector) 進行資料的徵集與挑選，尋找有用的電子資源。挑選後則轉交由採錄人員(Acquire)，若該出版品沒有版權的爭議，則直接交由編目人員(Cataloger)進行編目。若是有版權的問題，則待取得出版商/出版者/作者的授權後，才轉交由編目人員進行編目。在資料編日期間，隨時與相關部門(例如採編人員、資訊人員、資料挑選人員…等)交換所需資訊，資料徵選的最後決定權則交給採編人員決定是否要收錄該筆資料，逐步完成所有階段，最後將結果展示在使用者面前。

網路上的資源何其繁多，若說不具價值，那麼多數人還是習慣以搜尋引擎來搜尋網路上的相關資訊，那麼其意義何在？再者科技的進步，目前的 Blog、Wiki 等工具的普及，使用者可以很簡易的發表個人日誌、心得、文章...等到網際網路，通常使用者只需要具備上網的基本技巧，了解如何對文件進行編輯後，就可快速地將資料給呈現到網際網路上，當中的作者不乏有教授、知名作家、科技專精用戶...等專家，故網際網路上存在著許多優質的作品，是值得整理的。

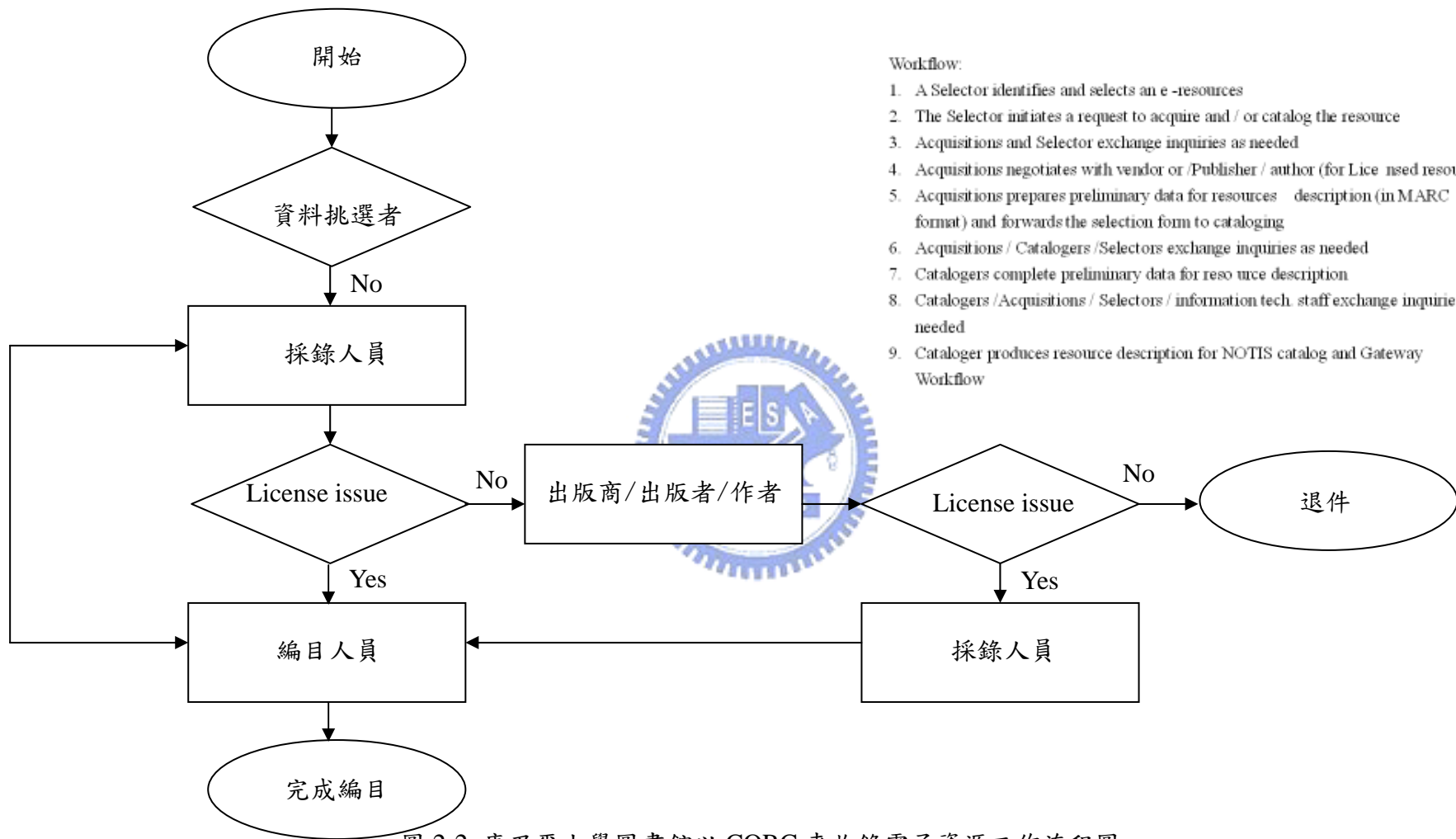


圖 2-2 康乃爾大學圖書館以 CORC 來收錄電子資源工作流程圖

參考來源：[21]

2.4 傳統主題式資源指引之問題研究

網際網路已成為主要的資訊傳播與獲取管道之一，但網路資源品質良莠不齊，圖書館多會另外建置主題式資源指引網站供讀者使用。R. Jackson and J. Pellack 研究報告指出，有 64% 比例的圖書館指出整理電子資源/主題資源指引是圖書館責無旁貸的責任；有 65-73% 的比例表示，圖書館所整理的主題式資源指引通常是可提供獨一無二的參考資源供讀者參考，各圖書館也積極建立相關主題指引供讀者使用[22]，但主題式資源指引網站的建置與後續的維護成本高，主題的分類可由單一主題到多主題的收錄，收錄範圍數量更是可多可少，收集的數量也以數千到數萬不等，這正也是主題式資源指引網站發展的瓶頸所在，收錄的量太少不足以吸引讀者使用，增加所收錄的量則建置單位人力與經費又無法負擔，透過各單位合作共建機制是折衷方式之一，但當中所面臨的問題不僅僅只有收錄的量，仍有其它相關問題存在，以下節次在於討論建立現行主題式資源指引網站所面臨的相關問題及解決之道，並透過喬治華盛頓大學調查報告來佐證圖書館所製作的主題式資源指引是具意義的。

2.4.1 主題式資源指引概況

90 年代起圖書館就積極的發展與建置主題式資源指引網站，主要是為了協助圖書館的使用者尋找可靠、經篩選的網路資源，例如國外所成立的 SOSIG、LII、IPL、RDNINFOMINE、Renardus、ASGF... 等；國內亦有國家圖書館所建置的「知識之窗：網路資源選介」。維護主題式資源指引網站資料的方式從最簡單的檔案管理，延伸成目前所使用的內容管理系統(Content Manage System)管理站內資料，這是目前的主流管理方式，例如 DESIRE 專案使用 RAOADS 軟體、George Washington University Gelman Library、Michigan University、Wayne State University Library System、Old Dominion University Libraries、Halle Library Eastern Michigan University 等圖書館使用 ResearchGuide 來管理及維護資源指引網站。

各主題式資源指引網站發展的目的多為協助特定使用者有關，故收錄的主題也依單位自行需求自訂，例如 SOSIG (The Social Science Information Gateway, <http://www.sosig.com>)，主要收錄社會科學、商業及法律主題、EdNA 與 GEM

則以教育界使用者為對象收錄與教學相關之主題、Academic Info 與 INFOMINE 則以學界人士為主，收錄範圍涵蓋生物學，農學，醫學，政府信息，地圖和地理信息系統，物理學，工程學，計算機科學，數學，社會科學和人文科學等。

主題式資源指引網站內容徵集與整理的方式，通常委以參考組(Reference services)、利用指導(Instruction services)及典藏部門 (Collection services) 等相關部門負責，而徵集主題的內容與範圍，有 66%的比例是透過主動要求全體教職員的協助達成，部份則是依據整理統計讀者所提出的需求而建置，圖書館主動要求協助的原因在於學科的範圍太過廣泛，而每人的專長都不全然相同，故理所當然資料的徵集，最快的方式就是請求校內現有的學科專家予以協助。讀者利用圖書館所提供的各式服務中不僅只是利用書目查詢系統來尋找書目，與使用線上電子資料庫與電子期刊服務，圖書館所另外建置的主題式資源指引網站，更是重要的一環，以 George Washington University Gelman Library 為例，該館所建置的主題式資源指引網站於一年內，總點擊統次數為 147,000 次 (統計時間為 2002 年 7 月 1 日 ~ 2003 年 7 月 30 日)。



2.4.2 主題式資源指引現行問題調查方式

問卷調查調查的施行方式可透過書面問卷、電子郵件與線上問卷調查的方式進行，R. Jackson and J. Pellack 所做的網路主題指引問卷調查(Internet Subject Guides Survey)即是透過傳統電子郵件的方式進行。進行線上問卷調查的好處在於，可以在不影響及加重現有的服務與不增加相關工作人員的負擔下進行，借助網際網路的線上特性，調查範圍更可涵蓋已有的主題式資源指引網頁。

喬治華盛頓大學即是採用線上問卷調查方式進行調查，問卷調查範圍不僅涵蓋已有的主題式資源指引網頁(調查當時，已建置各式主題達 89 種，另外 22 種主題為應因課程需求另外製作)，也能讓校外的使用者可以方便的進行填寫問卷，該校的實作方式一開始是採用彈跳式視窗來予以實作，但現況是有許多軟體會阻礙彈跳式視窗的跳出，例如 Google Tools Bar or Yahoo Tools Bar 都具有該功能。讀者可能會沒有發現圖書館正在進行這方面的調查，於是仿效 Amazon.com 的作法，將每個主題式資源指引頁面內直接嵌入檢查箱(check box)，可讓讀者可直接進行回應，線上問卷調查設計很簡潔，僅詢問

使用者對這份文件是不是有所幫助(很有幫助、有幫助、一點點幫助、沒有幫助)及看法(註釋的欄位)，如圖 2-3。藉由簡單的問卷設計從中調查並了解學生們的想法與意見。

圖 2-3 線上問卷調查表

資料來源：[6]

研究調查結果及討論整理如表 2-2，總計回收 210 份問卷，而網頁的總點擊數總計為 104,756 次，約只有 2%的回收率。由指引評分表中可以得知，52 % (非常有幫助 34.8%、有幫助 17.1%)的比例顯示，圖書館所製作的主题式資源指引的確對使用者有所助益。這份資料以科學數據來證實，不再只是單純的揣測。

表 2-2 指引評分表

Rating	Number of responses	Percentage
Very helpful	73	34.8
Somewhat helpful	36	17.1
A little helpful	17	8.1
Not helpful	68	32.4
Not rated	16	7.6
Total	210	100.0

資料來源：[6]

2.4.3 主题式資源指引現行問題與改善策略

建置主题式資源指引網站現行問題，可歸納成以下數點。

一、政策面議題：

1. 網路資源何處尋(Where)

2. 要由那些人尋找資源(Who)
3. 職員所需具備的能力(Staff and skill required)
4. 徵集策略(Selection policy)、收錄主題範圍、技術相關議題...等
5. 協力合作(Collaboration and cooperation)

二、技術面議題：

1. 互通性(Interoperability)
2. 採用標準(Standards compliance)
3. 多語言資源(Multilingual resources)

三、後續維護議題：

1. 斷掉的連結網址 (Broken links)
2. 關鍵詞搜尋(Keyword searches)/主題建議(Suggested topics)
3. 離題(Off-topic)
4. 描述資料過時、資源是否仍適用於當下
5. 部份圖書館與 OPAC 結合、網址更正時，需時時更正 OPAC 與主題式資源指引
網站內資料

就政策面相關問題，如網路資源何處尋、要由那些人尋找資源、職員所需具備的能力(Staff and skill required)、政策面議題...等，可透過良好的規劃、適當的教育訓練即可進行改善；就技術面而言如互通性(Interoperability)、系統若要交換彼此現有的檔案紀錄，有賴於雙方彼此遵循著共同的標準協定，詮釋資料標準可支援系統的互通性，提供豐富語義(semantic)。多語言資源(Multilingual resources)，截至目前為此，詮釋資料綱要(metadata schemas)，仍只針對單一語言環境設計，使用者可能需要其它的多語環境。採用標準(Standards compliance)，可確保系統資料的互通性(Interoperability)[23]。而主題式資源指引網站的後續的維護成本高是現行最大的問題所在，所建置好的網

站，往後仍是會面對資源連結的失效、描述資料的過時...等問題，錯誤難以立即獲得改善，R. Jackson and J. Pellack 研究報告亦指出，有 75%的比例顯示，認為圖書館建置主題式資源指引後續的維護是難以負荷的[22]。另一個較難解決的問題是 Reeb and Gibbons (2004)所指出的關鍵詞搜尋/主題建議，館員及學科專家所做的整理並不符合使用者的期望，這些分類讓讀者產生困惑，主要是因為資訊的提供者和讀者(學生)，彼此間的心智模型(Mental model)，不相同的緣故。讀者無法理解為什麼資訊的組織為什麼要如此分類，為什麼這些資源是歸屬在這個主題內。

喬治華盛頓大學圖書館針對建置主題式資源指引網站現行問題的改善方法方式如下：

1. 增加維護工作團隊：例如喬治華盛頓大學圖書館將原本維護人力，由原本的三個團隊成長到六個工作團隊，希望能更符合師生們的期許。
2. 網址失效(Broken links)問題：配合 Automatic link checking software)的輔助減少人工進行檢視，縮短連結檢查時間。人工進行檢視的工作流程分成六個步驟進行：
 - (1) 尋找失效網址連結。
 - (2) 工讀生回報給主題資源指引網頁的負責人(Guide owner)。
 - (3) 主題資源指引網頁的負責人，回報給網站管理人(Webmaster)。
 - (4) 網站管理人修改。
 - (5) 網站管理人修改完成再回報給主題資源指引網頁的負責人。
 - (6) 完成所有程序。
3. 關鍵詞搜尋/主題建議的問題：此類問題是難以獲得立即解決的，也許可以透過更為清楚的描述或是在每筆資源內重新定義它所呈現的標記。Reeb and Gibbons 的建議是為每個特定的課程建置資源指引，因為這樣可以讓學生們更直接、立即的了解要如何把相關的指引給串連起來。或是透與學生一起參與建置、共同合作的方式建置，建置出符合彼此 Mental model 的主題指引，不過

這樣的過程是一項長期且勞力密集的工作。當然由此模式所建置出來的模式，絕對更符合學生們所使用。

2.5 建置主題式資源索引的軟體

管理主題式資源網站清單的方法，可以區分成透過檔案管理及資料庫管理方式進行管理兩類。檔案管理的維護方式很簡單，清單的維護者只是很簡單地編輯文字檔案就能管理這些清單資源。而其運作的方式是採 Server-Side Includes 的方式把文字檔嵌入到 HTML 的頁面內。透過簡單的 Script 例如 `<!--#include virtual="abi.txt" -->`，就能把所維護的清單給呈現在使用者的面前[24]。資料庫管理的方式則是採用關聯式資料庫 (Relational Database Management System) 的方式來管理這些大量的清單資源。而以資料庫進行管理的軟體，比較著名的軟體有 ROADS、ResearchGuide... 等可供使用。由於 ROADS 專案已終止，故在本研究中實際要架設 Research Guide 軟體，並比較該系統功能，以做為開發系統之參考。

2.5.1 ROADS 系統

ROADS 系統 (<http://www.ukoln.ac.uk/metadata/roads/what/>)，可說是第一個以 GNU General Public License (GPL) 形式所釋出，專為管理主題式資源指引網站的內容管理系統軟體。該軟體開發的最主要是目的就是用來協助圖書館單位，提供一個簡單、容易使用的系統，共同建置及管理主題式資源管理網站。該系統自 2000 年 6 月釋出後陸續已有 SOSIG、Biz/ed、OMNI、ADAM、NOVAGate、EELS、The Finnish Virtual Library... 等 80 多所單位使用該軟體，建置開發主題式資源管理網站(資料統計時間至 2001 止)，[10]，甚至在美國大學的課程內，有些教授會專門把它介紹給研究生，將它視作講述電子圖書館課程內的一份教學文件。該軟體是以 Perl 語言撰寫。主要使用者介面是以 Web-based 的方式呈現。使用網際網路工程工作小組 (Internet Engineering Task Force, IETF) 所推動的 WHOIS++ Protocol 和 CIP (Common Indexing Protocol) 協定[19]。

該系統可以輕易地實現分散式資料庫的連結與檢索，可架設在任何使用 Linux Kernel 的作業系統上，該軟體的網站伺服器是採用 Apache Server Application。ROADS 的核心成員為 Martin Hamilton 及 Jon Knight，不過目前該專案已終止，相關資訊可由

SourceForge.net (<http://sourceforge.net/projects/roads/>)及官方網站上 (<http://www.ukoln.ac.uk/metadata/roads/>)獲得。

2.5.2 ResearchGuide 簡介

ResearchGuide 系統為密西根大學研究生 Kelsey Libner 與該校圖書館數位資訊服務館員 Suzanne Gray 共同建置，該官方網站為 <http://researchguide.sourceforge.net/>。主要開發的起源為密西根大學圖書館 (University of Michigan Graduate Library)，考量到每個圖書館館員對 HTML 的熟稔程度不盡相同，且要求所呈現的內容要以目錄結構的方式來呈現、收錄相關主題資源指引網頁，決定採用系統來協助圖書館資源指引網站的建立，藉以提供一個簡單的工具來協助該業務運作。簡單的說 ResearchGuide 主要是協助學術圖書館館員，以網路表單(Web Forms)的方式來建置主題指引和資訊頁面的開放原碼應用軟體。館員不需要理解及使用 HTML 的語法或是煩惱要如何使用 FTP 軟體來進行更新網頁，只要透過連結後端的管理介面，遵循著網頁表單所列欄位鍵入資料，即可將散亂各處的網路資源，整理成各種便於使用的主題指引。另外也可同時建置含聯絡資訊、學歷背景等訊息在內的學科專家背景介紹網頁，並藉由動態程式語言的協助，即時自動更新前端網頁內容[25]。

2.5.3 ResearchGuide 系統需求與功能

從技術面來探討，ResearchGuide 是採用 PHP 技術開發，後端採 MySQL 資料庫來儲存大量資料，前端則是以 HTML 網頁來呈現，符合 IEEE 的 POSIX (Portable Operating System Interface for Computer Environments) 標準，適合在各種 UNIX 平台上運行。有感於許多圖書館都面臨維護主題指引網站的挑戰，原軟體作者在徵得校方同意後，便讓 ResearchGuide 以開放原碼的方式，免費授權給需要的人使用，但 Kelsey Libner 後因工作忙碌，即使已規劃出將新增的功能，卻自 2002 年釋出 0.5 版的軟體後，便再也沒有發表任何新的版本。

ResearchGuide 軟體基本資料可參考表 2-3，該系統架構可從 Suzanne Gray 於 2003 年美國圖書館學會 (American Library Association) 年會展出的巨幅海報 (Poster)可獲得清楚的解答(圖 2-5)。

表 2-3 ResearchGuide 基本資料

軟體名稱：ResearchGuide
發表日期：2002-09-20
最新版本：0.5（2002-01-29）
授權方式：MIT License
作業系統：All POSIX (Linux/BSD/UNIX-like OSes), Solaris
程式語言：PHP
資料庫：MySQL
使用介面：Web-based
檔案下載： http://sourceforge.net/project/showfiles.php?group_id=63006
技術支援： researchguide-info@lists.sourceforge.net

資料來源：依官方網站上資料整理而得

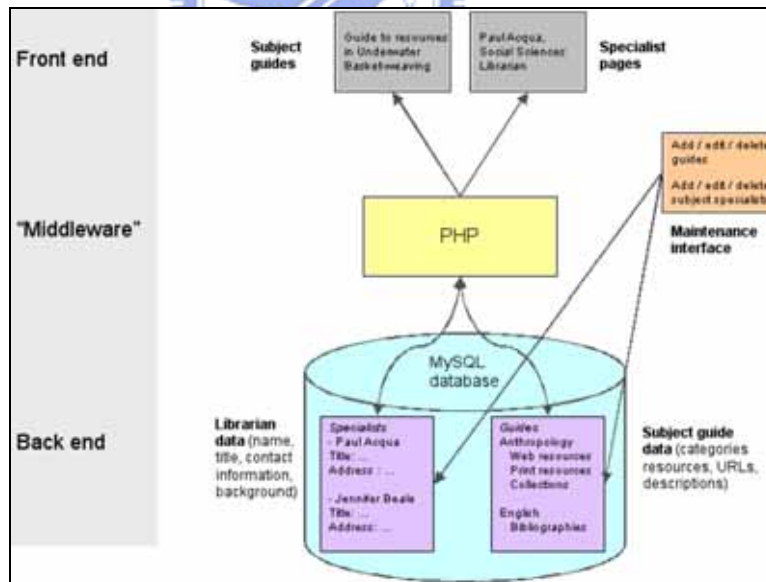


圖 2-4 ResearchGuide 系統架構

(資料來源：<http://researchguide.sourceforge.net/architecture.html> 上網日期：
2007/02/02)

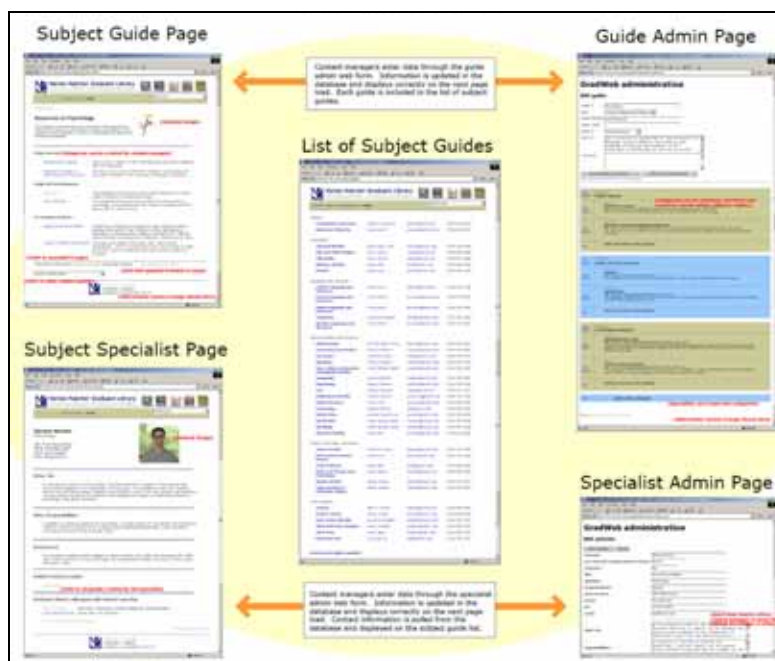


圖 2-5 ResearchGuide 網頁架構

(資料來源：<http://researchguide.sourceforge.net/architecture.html> 上網日期：
2007/02/02)

ResearchGuide 功能可區分為主題指引(a guide、all guides)、專家介紹(a specialist page、all specialist pages)及管理介面(Administration page)三類，以下為各項功能之介紹。

1. 主題指引(a guide、all guides)：

建置主題指引網頁提供一優良經選擇後的優質資源是 ResearchGuide 的主要目的，因此匯集所有主題指引而成的列表頁，即為其前端使用時的主要入口，列表頁可依領域 (area) 將主題指南分類，同時亦列出負責維護該主題指南的學科專家及其聯絡方式。每個學科皆為單一的主題指引，每份主題指引還可自行細分為參考工具書、文章索引及網路資源等多個類目，每個分類下可新增多項資源，每項資源皆可加上敘述及網址，以協助使用者瞭解或取得。

2. 專家介紹(a specialist page、all specialist pages)：

ResearchGuide 還提供建置學科專家介紹網頁的功能，網頁內包含其照片、聯絡資訊、職務說明及教育背景等個人資訊。任何由該學科專家所負責維護的主題指引，皆會自動顯示在此專頁中，以引導使用者連結到各主題指引去。而每份主題指引的頁尾皆會

自動顯示目前的維護者，以達到將各網頁串連起來的效果。另外，ResearchGuide 也會動態產生所有學科專家的列表頁，讓使用者更容易找到可請教的學科專家。

3. 管理介面(Administration page)：

管理介面的設計採用 Web-based，該功能為 ResearchGuide 之核心，採用網頁表單的形式設計。為避免各項設定及內容被使用者任意更動，管理介面進行權限控管是必要的，而 ResearchGuide 的使用權限可分為學科專家及網站管理者兩種層級。學科專家的權限較小，僅可針對自己所負責的主題指引進行編輯；網站管理者的權限較大，可對網站內的各個主題指引頁面進行新增、編輯或刪除的動作，也提供新增、修改或刪除學科專家的使用權限。如由校外連結管理介面，則一律需通過 ResearchGuide 的認證才可進入；如由校內連結，則因已通過校內的認證系統，因此無需再次認證即可進入管理介面。

2.5.4 ResearchGuide 應用實例

ResearchGuide 原是为密西根大學圖書館所設計，但在開放免費授權後，陸續有 Michigan University、Wayne State University Library System、Old Dominion University Libraries 及 Halle Library Eastern Michigan University 等圖書館用其建置主題指引，表 2-4 即为各校應用概況說明。



表 2-4 ResearchGuide 應用比較

學校名稱	Michigan University	Wayne State University Library System	Old Dominion University Libraries	Halle Library Eastern Michigan University
服務名稱	Subject Guides	Subject resources and guides	Subject Guides	research guide
上線時間	2002 年冬季	2003 年春季	2003 年 9 月 3 日	2003 年 12 月 3 日
指引分類數	6 類	未分類	8 類	11 類
主題指引數	43 份	51 份	38 份	74 份
資源分類	未統一	未統一	未統一	未統一
學科專家數	25 位	21 位	10 位	20 位
隱私聲明	未找到	有	有	未找到
圖書館數	32 所	6 所	---	---

資料來源：依各校網站上資料整理而得

2.5.5 ResearchGuide 測試與評估

為徹底瞭解以 ResearchGuide 與 Wiki 所建置主題指引之差異性，本研究實際進行安裝 Researchguide，該系統之測試環境如表 2-5，結果可由 <http://lib012.lib.nthu.edu.tw/ming/guide/> 進行瀏覽。安裝後需將原程式碼中的字元編碼由 ISO-8859-1 修改為 Big5，否則新增主題指引或學科專家時，輸入中文會出現亂碼。這主要是 Apache、MySQL 編碼問題所導致，故需額外調整。

表 2-5 ResearchGuide 測試環境

測試軟體：ResearchGuide 0.5
作業系統：Redhat 7.2
程式語言：PHP 4.2.3
資料庫：MySQL 3.23.58
網頁伺服器：Apache 1.3.0.3

內容的來源主要是來自國立交通大學 93 學年度下學期的教授指定參考書書單。內容的建置，則採用各學科依所屬學院分類，捨棄開課系所，直接以課程名稱取代各主題指引的學科名稱，讓使用者在瀏覽主題指引列表頁時，可更直覺選擇所需的主題指引。進入單一主題指引（圖 2-7）後，使用者將先看到課程說明或教學目標，協助其瞭解此份主題指引的用途；其次為此主題指引的相關資源，並細分為指定參考書、參考資料及相關連結等類目；頁尾的部分則顯示維護此主題指引的學科專家，也就是教授此課程的教授。盡可能讓每項資源包含連結，以便使用者快速取得，如為紙本資源，則連結到該資源的館藏目錄頁；如為電子資源，則連結到該資源的使用主頁；如校內無此館藏，則設法以其他公開的網路資源代替。



圖 2-6 主題指引列表頁

在學科專家介紹網頁內(圖 2-8)，則以學科專家的專長、聯絡方式及學經歷為主，並顯示由該學科專家所負責維護的主題指引。



圖 2-7 單一主題指引



圖 2-8 學科專家介紹

開放原碼軟體的優缺點已有諸多文獻討論，在此不贅述，僅就實際安裝 ResearchGuide 的經驗做分享。該系統之優點包含：

1. 採用 PHP 語言開發，該開發語言為目前主流之一，故只要有專業背景者皆可自行修改、開發，圖書館可輕易獲得技術支援。
2. 該軟體為 Web-based 設計，輸入資料採用 WebForm 方式進行，資料建置人員不需了解 HTML 或網頁製作軟體，即可輕鬆地建置主題指引網頁。
3. 內容與格式分開處理的架構，該架構可使多位圖書館員分工建置主題指引，但最後仍可擁有統一的網頁風格及內容樣式。
4. 各主題指引的細項分類預設為統一，但圖書館員仍可針對不同學科的特性，為主題指引分別制訂適合的資源分類，並可依分類的重要性自行編排順序，各主題指引間互不干擾。

系統之缺點為：

1. 該系統最大的缺憾為缺乏互動功能，管理者無法直接獲得使用者的主動回饋，此亦為以傳統 CMS 系統的先天限制。
2. 該系統主要的認證機制主要是以密西根大學所設計，對想採用此系統的單位無法立即直接套用，仍需額外修改認證模組功能。

3. 新增、指派主題指引，為網站管理者才有的權限，因此學科專家僅能被動等待使用權限的增加，才能對主題指引進行編輯。
4. 系統缺乏統計功能：「大學圖書館設立及營運基準」第 35 條明確指出「大學圖書館應定期進行館藏、讀者服務及技術服務之調查統計，實施績效評估，並據以改進服務品質[26]。ResearchGuide 並無相關統計報表可供參考，難以得知目前網站內已建置多少有參考資訊或讀者的使用狀況。

2.6 Web 2.0 與 Wiki

2.6.1 Web 2.0

Web 2.0 的定義眾說紛紜，有些人說是讀 / 寫的網頁(Read/Write WEB)，有些人說是參與式的網際網路 (Participatory Web)，也有些人說它是使用者為導向的網頁 (User-Initiated Web)、現場直播網頁(Live Web)。不管說法如何，Web 2.0 改變了傳統的溝通方式，提供一個展新的傳播媒介，資訊傳播從傳統的方式單向進行，演變成可雙向溝通模式發展，提供了展新的契機，可讓不同的人員共同參與其中，讓使用者可進行簡單的溝通、互相參與、分享、創造及線上出版資訊的基礎平台，以目前而言，Web 2.0 最受歡迎的工具軟體，就是大眾耳熟能詳的部落格 (Blog) 及 Wiki 引擎 (Wiki Engine)[27, 28]。

「Web 2.0」這概念，一開始是從 O'Reilly 與 MediaLive 討論國際研討會議題開始的。Dale Dougherty 這位網路先驅者，同時也是 O'Reilly 公司的副總經理，注意到一個事實，自從網路崩盤後，許多重要、精彩，極受歡迎的網路新應用，突然有規律地出現。從網路崩盤中倖存的網路公司，似乎都有一些共同點，這是否意味網路崩盤正代表重要的網路轉折點，或者可以把這些新應用的特徵，統稱為「Web 2.0」呢？Web 2.0 的網站特徵已經被廣泛運用，而且網路上也出現許多 Web 2.0 的線上研討網站[29]。簡言之，Web 2.0 並不是一個具體的事物，而只是一個階段，為了讓這個階段能成功運作，所組合而成的各種服務總稱。若僅由技術面來探討的話，只要網站服務內有採用到以下相關技術者，均可宣稱是屬於以 Web 2.0 為核心的網站 Web 2.0 網站[30]。

- 豐富網際網路的應用程式，例如使用 AJAX 技術。

- 網站採用 CSS(Cascading Style Sheets)技術。
- 採用語義化有效的 XHTML 標籤，例如使用 Microformats。
- 採用 RSS/ATOM 數據的聚合技術。
- 清楚有意義的網址(URLs)。
- 採用大眾分類法(Folksonomy)讓使用者對資源自行定義標籤(tag)。
- 採用 Wiki 軟體。
- 採用開放源始碼軟體(Open source software)，例如使用 LAMP 架構網站。
- 採用 Weblog 軟體出版發行。
- 採用混搭(Mashups)的機制，豐富網站內容，通常都是透過開放 API、Web feeds (如 RSS 或 Atom) 、以及 JavaScript 的技術達成。
- REST 或者是 XML 網站服務應用程序編程介面(XML Webservice APIs)。

J. Cuene 將傳統的 Web 與 Web 2.0 二者之間的差異性整理出下表的資料，從表 2-6 可以看出來，Web 2.0 比之於以前，容許更多使用者參與網路內容的產生，使得網路內容不再侷限在懂得 HTML 或是具備網路上下傳使用能力的少數玩家手上，而成為人人均可以創作的園地，也因如此，互動性在 Web 2.0 時代裡更被強調，不論文字或是多媒體內容均是如此，這也該是 Web 2.0 的核心精神[31]。

表 2-6 Web1.0 與 Web 2.0 觀念比較表

	Web 1.0	Web 2.0
時間	1993-2003	2003-
表現方式	透過瀏覽器觀看大量網頁	網頁可引入其它網站的內容、加強互動性，比起傳統的網頁，更像一個應用程式。
模式	讀	寫與貢獻
主要的內容單元	頁面	發表/記錄
瀏覽方式	瀏覽器	瀏覽器、RSS 閱讀器、其它
架構	Client Server	Web Services
主導者	電腦玩家(geeks)	大量的業餘人士

(資料來源：http://cuene.typepad.com/blog/2005/05/web_20_at_mima.html 上網日期：
2006/11/23)

在中文版維基百科也定義 Web 2.0 的通用參考概念如下：

1. 網站不能是封閉的：網站必須可以很方便地被其他系統獲取或寫入數據。
2. 用戶應該在網站上擁有用戶本身所擁有的數據。
3. 完全地基於 Web：大多數成功的 Web 2.0 網站可以幾乎完全通過瀏覽器來使用

由 Web 2.0 Directory 可以整理出表 2-7，藉以觀察並了解 Web 2.0 目前所應用的概況[32]。Keven 更預估 Web 2.0 的應用，是一種蔓延趨勢，可由點及面，由娛樂而商務，由工具而平台，由惠及個人而顛覆各行各業。[33] 也就是說目前 Web 2.0 的應用可以廣泛應用且延伸到各個產業。

表 2-7 Web 2.0 網站類型統計

項目	數量	說明
過濾服務 (Filtering Services)	184	進行內容/新聞/Feeds 的過濾服務；允許使用者選擇內容..等
協同合作服務 (Collaboration Services)	125	提供用戶分群，進行合作/分享資訊，例如 Blog、Bookmarking、tasks 等。
對等服務 (Peer Services)	114	提供用戶評價、評論、排序等共同合作組織資訊的服務
搜尋服務 (Search Services)	110	提供對於文件、音樂、Blog 內容、影音頻等輸入搜索的服務
部落格服務 (Blogging Services)	106	提供 Blog 註冊及相關功能的服務
相片服務(Photography Services)	92	提供線上儲存、分享、編輯、管理、列印相片與圖片 ...服務
RSS Services	90	提供線上閱讀、管理 Atom/Rss feed Aggregation 的服務
商業服務 (Business Services)	75	提供企業導向各類 Web 2.0 服務
編碼服務 (Programming Services)	73	提供程式編寫、展示、概念證明、程式碼共享等服務
電子商務服務 (eCommerce Services)	64	提供電子商務解決方案，例如販賣商品、協助購買物品等服務
影音服務 (Video Services)	64	提供影音的製作、發掘、搜尋、共享、儲存服務
線上書籤服務 (Bookmarking Services)	54	提供線上書籤的儲存、搜尋、共享

項目	數量	說明
預告服務 (Coming Soon! Services)	51	提供「預告」服務
地圖服務 (Mapping Services)	50	提供地圖服務，例如使用 Google Maps
音樂服務 (Music Services)	47	提供創作、發掘、搜索、分享、儲存音樂的線上服務
辦公軟體 (Office Services)	45	辦公軟體的線上版服務，可取代現有的辦公軟體
網路服務 (Networking Services)	40	允許使用者進行線上連線，共同創造社會性網路(social networks)
Podcasts Services	36	提供創造、管理、搜尋 podcasts 服務
行動電話服務 (Mobile Services)	33	提供使用者行動電話服務
消息服務 (Messaging Services)	32	提供各種消息傳遞客戶端服務，與聊天服務有所重疊
電子郵件服務 (Email Services)	30	提供電子郵件(包括聲音郵件)等服務
儲存服務 (Storage Services)	30	提供線上的檔案儲存及傳輸服務
目錄服務 (Cataloging Services)	26	提供用使用者分享圖書、CD、DVD 等目錄列表、評價的服務
旅遊服務 (Travel Services)	24	提供旅遊/旅館資訊共享，訂票服務等
聊天服務 (Chat Services)	23	提供線上進行聊天服務，有些會與消息服務有所重疊
個人首頁服務 (Start Pages Services)	23	提供客製化個人首頁服務
本地化服務 (Local Services)	22	本地化社群的相關服務；部份和社群服務重複
月曆服務 (Calendar Services)	21	提供網絡月曆及行程提醒、計劃提醒等功能
混搭服務 (Mashup Services)	20	網站可混搭其它 Web 2.0 網站的 API，例如 amazon, blogger, del.icio.us, flickr 等站所提供的功能
Framework Services	19	Platforms / frameworks to run other software solutions etc
瀏覽服務 (Browsing Services)	18	提供網站瀏覽器及外卦程式等
Financial Services	18	Sites dealing with personal finances / money / networth etc
社群服務 (Community Services)	17	分享社群所建置的資料；部份與本地化服務重疊
事件服務 (Events Services)	17	當地重要事件的服務；與日曆服務有所重疊
Other Lists	17	提供參觀 Web 2.0 Blogs/列表(Other Web 2.0 blogs/lists to visit)
記事服務 (Note Taking Services)	16	線上提供記事的服務；部份服務與 Wiki 服務重疊

項目	數量	說明
任務列表服務 (To Do Services)	14	線上排定工作列表，部分功能與日曆服務重疊
職業服務 (Employment Services)	13	提供工作、換工作、長期/短期的工作服務
串流服務 (Streaming Services)	13	線上多媒體影音串流的服務
網路分析服務 (Web Analytics Services)	12	協助調查 Blog/網站 的使用率調查等服務
維基服務 (Wiki Services)	10	線上社群、共同編輯、發佈、分享資訊的服務
遊戲服務 (Game Services)	9	與遊戲相關的服務
位置跟蹤服務 (Geotracking Services)	9	跟蹤使用者的 IP/位址，與手機服務部份有所重疊
無盈利服務 (Non-Profits Services)	9	非營利組織單位所成立的服務
Grassroots Services	7	當地化服務；在地人幫助其它外人者
幽默服務 (Humor Services)	6	好玩有趣幽默的服務
出版服務 (Publishing Services)	6	圖書 / 列印 / 即時出版服務(on-demand publishing)
投票服務 (Polling Services)	5	投票 / 調查 服務
專案管理服務 (Project Management Services)	5	協助設計大型專案的執行
分散合作的服務 (Cooperative Distribution Services)	4	檔案文件的交換採用類似於 BT 的方式進行
設計服務 (Designing Services)	4	提供色彩協調及設計的服務
科技支援服務 (Tech Support Services)	4	提供參考諮詢、軟體技術等
影音 Blog 服務 (Vblogging Services)	4	影音 Blog 的服務
無線服務(Wi-Fi Services)	2	提供無線資訊/分享的服務
邀請服務 (Invitation Services)	1	提供線上邀請服務

(資料來源：<http://www.econsultant.com/web2/>。上網日期：2006/11/27)

表 2-6 及表 2-7，整理出 Web 2.0 的通用概念及目前應用概況，研究者認為 Web 2.0 根據的最重要特性整理如下：

1. 服務是以 Web 為基礎平台，網站內的資訊是開放的，使用者可以很方便、簡易的使用/寫入資訊。
2. 工具的使用單純化，使用者不再需要藉由 WYSIWYG (What You See Is What You Get) 的軟體來協助撰寫 HTML 文件，通常需具備基本的上網操作技巧。藉由瀏覽器(如 IE 或 Firefox)就能參與平台間的互動。
3. 資訊的流向不再是單方面的推進，而是朝著雙向溝通的方式改變，也就是說資訊的提供，也不再是少數人所主導，而是任何想參與的使用者都可以自由的貢獻本身的經驗與想法。
4. 「互動」與「分享」，利用群體的智慧彼此分享經驗與想法，共同參與主題內容的創建。

2.6.2 Wiki

從維基百科中對 Wiki 的定義，可以得知 Wiki 一詞，是源自夏威夷語的「wee kee wee kee」，為「快點快點」之意。Wiki 系統指的是一種可在網路上開放多人協同創作的超文本系統，Wiki 是由「Wiki 之父」W. Cunningham 於 1995 年所創，主要是由軟體設計模式社群發展出來，發展之初主要是用來書寫與討論模式語言。W. Cunningham 於 1995 年 3 月 25 日成立了第一個 Wiki：WikiWikiWeb，用來補充他自己經營的軟體設計模式網站。Ward 發明了 Wiki 這個名字以及相關概念，並且實作了第一個 Wiki 引擎。W. Cunningham 說 Wiki 一詞是根據檀香山的 Wiki Wiki 公車命名而來如圖 2-9，這是到檀香山學會的第一個夏威夷語，Wiki 的構想主要是來自 1980 年代晚期利用蘋果電腦 HyperCard 程式作出的一個小功能[34]。

Wiki 是一種基於 Web 的軟體，它允許使用者藉由瀏覽器改變線上所看到的頁面內容。這讓 Wiki 的使用者，可以藉由 Wiki 平台十分簡單且容易的集體合作在同一主題及這些超文件上。[35]即便是沒有網站設計體驗的使用者，也可以合作編寫網站。任何在同一社群的成員，都可以自由的新增、編輯同一份工作。當然它是一種迷人的工具，它可以在任何時間，讓使用者可以直接在網頁上直接寫下其想法與意見。大部份的操作者在第一次使用 Wiki 時，都十分的驚訝，竟然能夠直接在網頁上修改內容。



圖 2-9 檀香山的 Wiki Wiki 站牌

(資料來源：<http://en.Wikipedia.org/Wiki/Wiki> 上網日期：2006/02/02)

換句話說 Wiki 是一套可以簡易創造、改變 HTML 網頁的系統，再加上可記錄所有的更動歷程，並可為每個條目定義其分類(標籤)的系統，系統同時也提供隨時可還原等機制功能...等等。使用 Wiki 系統所架設的網站可稱為 Wiki 網站，Wiki 網站容許任何造訪的人，可以能快速輕易地加入、刪除、編輯所有的內容，甚至連身份的認證都不需要，它特別適合團隊合作的寫作。Wiki 系統包含許多輔助工具，可讓使用者可以輕易追蹤 Wiki 文件的變化歷程，透過”討論”的功能可以直接進行商討，Wiki 所產生的內容也許有誤，因為使用者必定會加上不正確的資料，但是當下個使用者發現時，就可立即的線上即時更正錯誤，任何人都可以隨時回復到之前的版本，以求最為正確、高品質的內容。目前採用 Wiki 最為成功的案例是維基百科(Wikipedia)，根據美國亞馬遜旗下 Alexa (全球網站流量統計評比公司) 統計，Wikipedia 每個月約有 50 億的頁面瀏覽率，全球排名約在前 20 名左右。當然 Wikipedia 的瀏覽率高，背後一定也有著豐富的資料規模，目前英文 Wikipedia 就擁有 10 億單字的內容，約一百多萬篇文章，所含的內容已經超過紙本的大英百科全書。而德文 Wikipedia 緊追在後，擁有 38 萬篇文章，約與紙本德文百科全書的文章數量相當。中文 Wikipedia 方面，則有六萬二千五百餘篇，其他如法國、日本也有 20 多萬篇文章。在 Wikipedia 上面有著 200 多種不同的語言，總文章超過 350 萬篇。且超過一萬篇文章有著 38 種語言版本，超過一千篇文章有 75 種語言版本。WikiPedia 背後所為它建置龐大資料撰寫條目及內容的維基人，多為二十

至三十歲的年輕社群，J.Wales 表示四十至五十歲的維基人相對來說少很多，可能的原因是年紀較大的使用者對於上網行為較不熱衷，或是所需的技術門檻較高的緣故。維基人的參與編寫狀況的統計呈現一個有趣的現象：約有 50% 的人參與編寫了 0.7% 的內容，而 1.9% 的人參與編寫了 72.8% 的內容。這顯示 Wikipedia 主要是靠著一小群高度熱情參與的人士，大量地在持續提供內容，分佈在世界各處的維基人創造了 350 多萬篇文章，但實際上該專案所投入的人力，僅有 2.5 個成員，其中一個是管理網站平台、維護伺服器主機的技術人員，另一位是 J.Wales 的助理，0.5 個則是仍未畢業的研究生 [36]。

也有人認為，Wiki 系統屬於一種人類知識的網路系統，人們可以在 Web 的基礎上對 Wiki 文本進行瀏覽、創建、更改。而這種創建、更改、及發佈的代價遠比 HTML 文本小，可由表 2-16 中得知，若採 Wiki 的機制運作，確實比一般的方式更為快速且成本低；與此同時 Wiki 系統也支持那些面向社群的協作式寫作，為協作式寫作提供了必要的幫助；最後，Wiki 的寫作者自然構成了一個社群，Wiki 系統為這個社群提供了簡單的交流工具。與其它超文本系統相比，Wiki 有使用簡便且開放的特點，所以 Wiki 系統可以幫助人們在一個社群內共享某個領域的知識 [34]。也就是說它可以取代原有的網頁發布方式，資訊的提供，不再需要特定的專業人士，如伺服器管理者或網頁編輯師來專門負責撰寫 HTML 文件。網頁的發布不再需要透過 FTP (File Transfer Protocol) 上傳到網站伺服器上。只要簡單的流程，透過 Wiki 語法，就能完成資訊的發布、修改。資料的錯誤可以獲得迅速的改善，改變原有資料錯誤被使用者指正的窘境。使用者發現網頁錯誤必須通報網站的連絡人，連絡人在通知伺服器管理人修改，此時使用者才能得到正確的資訊。網頁的維護與使用者的交流方式，也不再是以單一 Push 的方式進行。只要有興趣的使用者隨時可參與，協助特定主題的內容創作。這就是 Wiki。

2.6.3 Wiki Engine

Wiki 引擎 (Wiki Engine)，或稱為 Wiki 軟體 (Wiki Software)，是指用來架設 Wiki 系統的軟體。廣義來說，即是一種軟體能作為網路共筆，供網友自行編輯，並最終集成完整的資料庫，狹義來說，即是能達成維基百科樣式的軟體 [37]。Wiki Wiki Web 是第一個使用 Wiki 所成立的站點，它於 1995 年 3 月 25 日成立，原本是 Portland PatternRepository 波特蘭樣式語言社群額外使用的自動化工具，從該網站可以了 Wiki

的發展歷史，及目前所流行的版本有那些 Wiki 引擎可供採用[38] 並可由此統計出目前所流行的 Wiki 引擎總計有 270 種，而所開發的程式語言更達 39 種之多。

序號	開發語言	總計	序號	開發語言	總計
01	PhpLanguage (PHP)	75	21	AwkLanguage	2
02	PerlLanguage	43	22	VisualFoxPro	1
03	JavaLanguage (Java)	34	23	Vim Script	1
04	PythonLanguage	29	24	SuneidoPlatform	1
05	RubyLanguage	12	25	Rexx	1
06	SmalltalkLanguage	9	26	PrologLanguage	1
07	CsharpLanguage (C#)	8	27	newLISP	1
08	ASP	6	28	MILanguage (ML)	1
09	SchemeLanguage	5	29	LotusNotes/Domino	1
10	VisualBasic	4	30	JavaScript	1
11	CeePlusPlus (C++)	4	31	IoLanguage (Io)	1
12	CeeLanguage (C)	4	32	Ibasic	1
13	Unknown	3	33	HtagLanguage	1
14	ToolCommandLanguage (Tcl)	2	34	Haskell	1
15	RebolLanguage	2	35	ErlangLanguage	1
16	ObjectiveCaml	2	36	EasyLanguage	1
17	LuaLanguage	2	37	CommonLisp	1
18	EmacsLisp	2	38	Cheshire	1
19	ColdFusion	2	39	AdaLanguage	1
20	Bash	2		Total	270

表 2-8 Wiki 引擎開發程式語言統計表

(資料來源：<http://c2.com/cgi/Wiki?WikiEngines>。上網日期：2006/09/27)

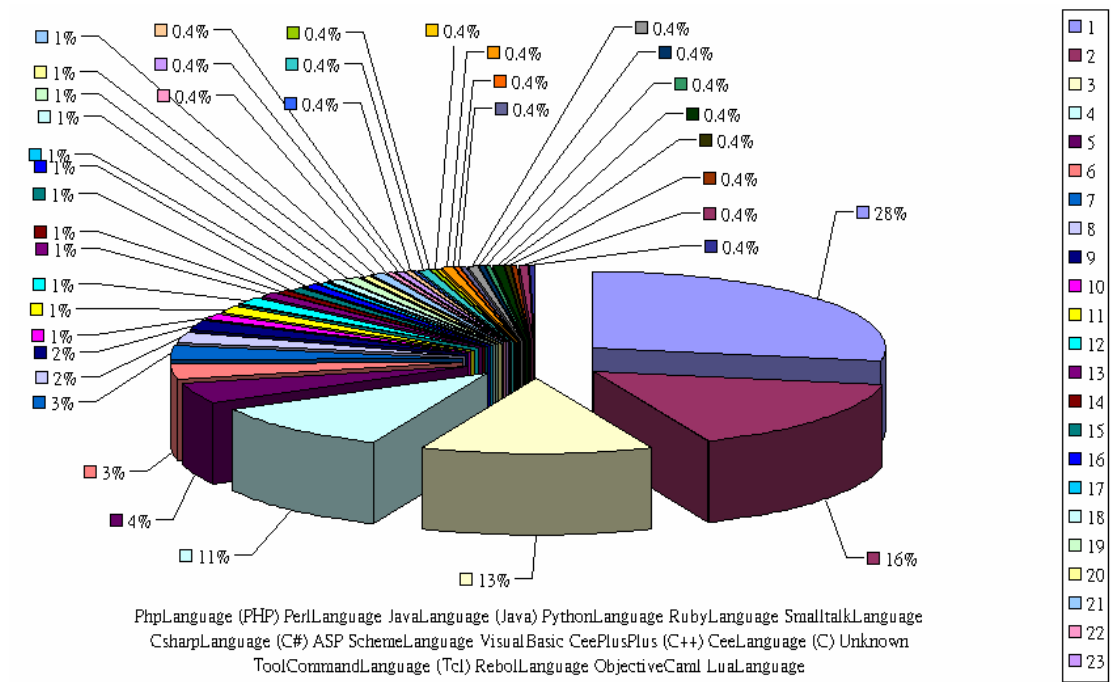


圖 2-10 Wiki Engine 開發程式語言比例圖

由表 2-8 及圖 2-10 可發現，目前 Wiki Engine 多採 PHP、Perl 與 Java 等語言來做為開發的主要工具，這和跨平台性及普遍性有關，因為不管是以 PHP 或 Java 開發的系統，大多可以運行在 Linux-like 平台及 Windows 平台上。

大多數的 Wiki 引擎採用資料庫的方式來保存其資料，也有少數是選擇以檔案的形式來保存資料，以 PHP 開發的 Wiki Engine 其後端所選用的資料庫大多採 MySQL 或 PostgreSQL。

Wiki 引擎的挑選因素，因人而異。W. Cunningham 認為，在選擇 Wiki Engine 時有三個因素特別重要：

1. 區別建置的目的：不同的應用目的，就會有不同的需求、不同的工作流程產生，選用一個合適的系統，不僅可減少開發人員發展程式碼(Coding)的過程，也避免了成本的浪費。例如說，若比較注重的是管理的方便性，文件的修改歷程都需要紀錄備查，這時就需要考慮到 Wiki 系統是否已有歷程記錄等相關功能。
2. 軟體突顯本身特色：

有些 Wiki 強調安裝容易、有些則強調使用者界面的設計、有些是提供討論的區域可讓參與者直接進行討論決定一同編寫的方式。目前有各式開發語言所撰寫的 Wiki

系統。Wiki Syntax 目前也未統一，有些 Wiki Syntax 的設計更為直覺，容易接受例如使用”|”符號取代原有 HTML <TD><TR> 的表格標籤寫法，更為直覺，這些因素當然也影響著 Wiki 的挑選。

3. 較多人下載：

較多人的下載，表示著使用者對它的接受度及肯定。愈多人的使用，能讓系統更加完善。以 MediaWiki 為例，自 2005 年版本的修正由 1.5.rc3，陸續的修正到目前的 1.9.3 版本，當中不僅針對安全性問題(例如 XSS 漏洞…)作修正，也陸續地新增使用者所需要的各項功能。

W. Cunningham 在其 Wiki Wiki Web 內，從各功能性進行分析整理，列出當下前十大 Wiki 引擎 如表 2-9 所示

表 2-9 排名前十大的 Wiki 引擎

序號	Wiki 引擎	開發語言	備註說明
01	MoinMoin	Python	使用 Python 程式語言開發，通常彈性與模組式(modular design)的設計是系統最大的特色
02	MediaWiki	PHP	為 Wikipedia 的專案所採用的平台，為目前最為廣泛所採用的 Wiki 軟體，開發程式語言為 PHP，資料庫則採用 MySQL
03	PhpWiki	PHP	採用 PHP 語言開發，由原 UseModWiki 改版也新增許多新功能 (A very popular PhpLanguage Wiki based on UseModWiki, with many features added.)
04	OddMuseWiki	Perl	依據 UseModWiki 的特性予以修改，採用採用 Perl 做為開發語言
05	UseModWiki	Perl	A PerlLanguage Wiki, based on Ward's original WikiWiki. 採用 Perl 程式語言開發，該 Wiki 是源自 W. Cunningham 所開發的 WikiWiki
06	TWiki (TWikiClone)	Perl	A powerful, skinnable, extensive PerlLanguage Wiki, aimed at large corporate Intranets. 軟件定位為「靈活、強大、易於使用的企業協作平台」，運行於 Perl 環境，該 Wiki 不採用資料庫
07	TikiWiki	PHP	A has-everything content management system with a powerful Wiki (PHP). 非常優秀的網站內容管理系統，採用 PHP、ADODB、Smarty 等技術構建，功能齊全
08	PmWiki	PHP	採用 PHP 程式語言開發，資料的儲存不採用

			資料庫形式、容易安裝、風格清爽是它的特色
09	Best akkaWiki fork	PHP	採用 PHP 程式語言開發/使用 MySQL 資料庫

(資料來源：<http://c2.com/cgi/Wiki?TopTenWikiEngines>。上網日期：2006/09/28)

2.6.3.1 TiddlyWiki Engine

在所有 Wiki 引擎中，TiddlyWiki 頗為特殊，因為它是屬於 Client Side Script，完全採用 HTML、Javascript 及 CSS 為開發語言，使用者可很簡單且直覺的使用，而且最特殊的一點是，該 Wiki 完全不需使用到任何的資料庫，當然也就不需要另外架設網站伺服器。僅需要擁有 TiddlyWiki 的檔案，隨時隨地就可編輯它，而且僅需要瀏覽器就能夠完成所有的編輯動作。無論是 Firefox、IE 都支援 TiddlyWiki 的操作。

TiddlyWiki 是由 J. Rustone 所開發的，原始的開發目的是主要是用來做為個人使用的筆記本。TiddlyWiki 是一種新穎的媒介，讓寫作者得以專注於文字內容及寫作風格上。它其實有點類似 Blog，因為它把內文劃分成整潔的小的文章段落(Tiddlers)。但它的閱讀方式不像在閱讀書本，需要全文的參閱。它就像在看百科全書一樣，不理解的內容，再藉由著超鏈接的方式串聯彼此相關內容。有點類似非線性的 Blog 各自發揮內容、分散於各方的文章段落(micro content)，然後在彼此凝聚成一致性線性 Blog 聚合物。

Tiddly 本身也擁有自己的發展社群，不管是在使用上、臭蟲(Bug)修正...等等之類的問題，都可以在下列網站中，尋求到幫助[39]。

- Google Group：TiddlyWiki General discussions
- Google Group：TiddlyWiki Development Developer's discussions
- Google Group：GTD TiddlyWiki Applying TiddlyWiki for GTD

由 Google Trend 如圖 2-11，可以觀察到 TiddlyWiki 的使用率是愈來愈來普及。

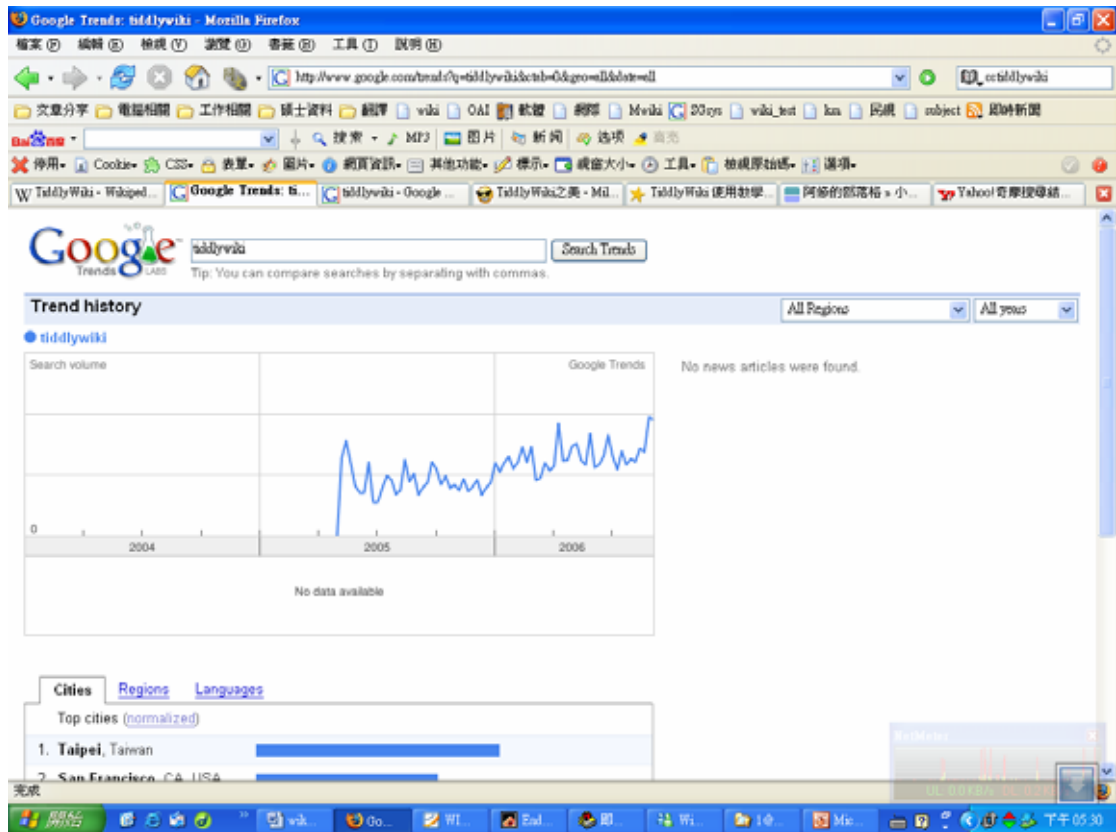


圖 2-11 TiddlyWiki 在 Google Trend 曲線圖

(資料來源：<http://www.google.com/trends>。上網日期 2006/9/27)

2.6.3.2 ccTiddly

如前所述 TiddlyWiki 屬於 Client side script，也就是說它不具有 Server / Client 的架構，因此 TiddlyWiki 的系統架構設計僅適合個人使用，無法達到共筆的功能。而 ccTiddly 是眾多 Wiki 軟體的後起之秀，第一次釋出時間是在 2006 年 1 月，原始作者為 CoolCold，創作的主要原因是由於 TiddlyWiki 的架構採用 Client-Client 架構，不適合應用於共同創作編輯的應用，CoolCold 在使用 TiddlyWiki 時總是需要透過隨身碟、FTP 或 E-Mail 的方式來存取檔案，也希望能讓有興趣參與的使用者能共同編輯 TiddlyWiki，檔案交換過 FTP、E-Mail 的方式總是不理想。CoolCold 重新以 PHP 語言撰寫 Server Side 的版本，於是 ccTiddly 就這麼誕生了[40]。

2.6.3.3 ccTiddly 的系統需求及系統安裝

- 系統需求

依官方網站的建議，只要有連接網路具有 HTTP server (Apache 2.0 / Apache 1.3) 、PHP4 / 5 的環境，後端的資料庫為 MySQL 即可安裝，資料庫版本的選擇建議採用 4.1 以上的版本，編碼方式採用 UTF-8。

- 系統安裝

ccTiddly 設定檔 (/config/default.php)-主要是設定與資料庫連接的一些參數值

說明：第 2 到第 6 行，主要是設定與資料庫連接的一些參數，在這裡需要指定 MySQL 的使用帳號及密碼，若 MySQL 是採用不同的埠號，也可在 sql host (第三行的地方)，指定所需使用的埠號。這是因為同一主機內，可能其它軟體，已經先行佔用了 MySQL Default Port 3306 這個埠號了，此時就需另外指定連接的埠號位置。

其它的相關參數，如語系設定檔(官方只提供英文語系檔)，或是有些頁面限制使用者刪除或修改、cookie 的存活時間、鎖定頁面不讓人刪除，或是有些頁面需要登入後才會呈現出來等等的設定，都可以在 /config/default.php 做相關參數的調整。

```
<?
// database/table
$tiddlyCfg['db']['host'] = "localhost";          //sql host
$tiddlyCfg['db']['login'] = "your login account"; //login name
$tiddlyCfg['db']['pass'] = "your login password"; //login password
$tiddlyCfg['db']['name'] = "your db name";       //db name
$tiddlyCfg['table']['pref'] = "";               //table prefix
$tiddlyCfg['table']['name'] = "your table";
$tiddlyCfg['table']['backup'] = "yourbktable";

// site preferences
$tiddlyCfg['pref']['tw_ver'] = "tw_ok_tiddlyWiki2.0.10"; // choose between different
version of TW, or adaptation
$tiddlyCfg['pref']['language'] = "en"; // choose between different version of TW, or
```

```

$StiddlyCfg['pref']['delete'] = 1; // 0 = disable deleting, useful to find previous versions
of a tiddler if the tiddly was edited by unauthorized user

$StiddlyCfg['pref']['version'] = 1; // 0 = no versions stored, 1 = all versions stored. The
version number is always updated

$StiddlyCfg['pref']['cookies'] = 20; //cookies expire time, in minutes [0=disable]

$StiddlyCfg['pref']['lock_titles'] = array(); //lock titles

....

?>

```

圖 2-12 ccTiddly v0.5.5 的 default.php 內的部份程式碼

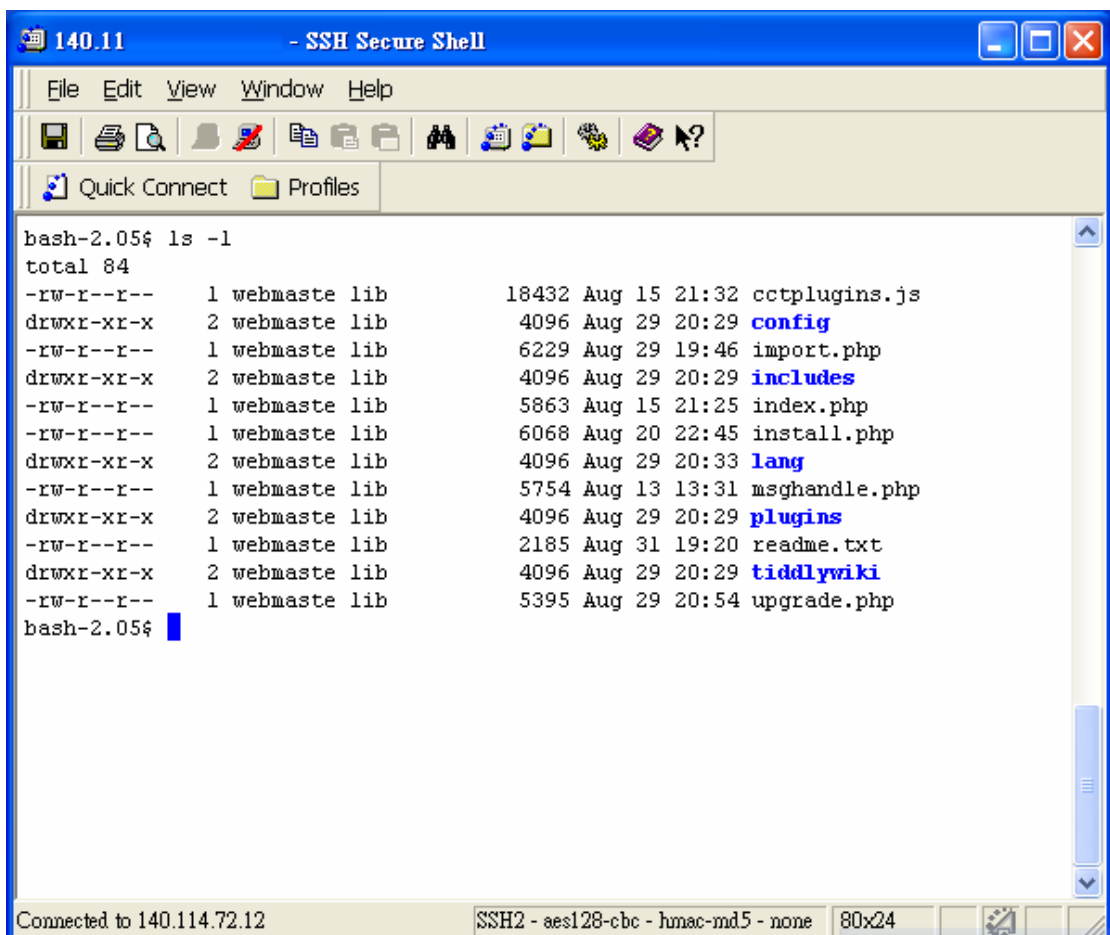


圖 2-13 ccTiddly 的目錄結構圖

該系統預設的檔案結構及相關功能如表 2-10、2-11 與 2-12。系統安裝完成後，在資料庫內會產生兩個資料表(tiddly_Wiki_entry 與 tiddly_Wiki_entry_version)，tiddly_Wiki_entry 資料表主要是作為儲存內文(Wiki Page Script)，而

tiddly_Wiki_entry_version 資料表則是作為版本控制的目的而存在。系統的安裝簡單，且介面清爽，和其它的 Wiki 引擎 有很大不同的差異。也許 ccTiddly 它的功能面不似 MediaWiki 或 TikiWiki 那麼完整，但基本上 Wiki 所需要的基本功能，ccTiddlywiki 皆有。

表 2-10 ccTiddly 系統檔案結構及功能

項次	目錄	檔案名稱	功用描述
1	config	default.php	基本的設定檔
2	includes	Config.php	引入 config 中設定，並設定使用何種語系程式檔
		functions.php	定義程式檔所需使用之函數
		print.php	將資料庫內的資料，顯示在網頁上
3	lang	en.php	語系支援檔 (英文)
4	tiddlyWiki	tiddlyWiki2.0.10.js	主程式檔
5	index.php		首頁，會呼叫程式檔

表 2-11 資料庫表格 tiddly_Wiki_entry

表格名稱	tiddly_Wiki_entry	說明	ccTiddly 之資料表，記錄 Wiki 內所有資訊			
欄位名稱	型態	大小	主鍵	校對	NULL	說明
id	int	11	Y	utf8_general_ci	N	auto_increment
title	varchar	255	N	utf8_general_ci	N	Wiki 標題
body	text		N	utf8_general_ci	N	標題內文描述
modified	varchar	128	N	utf8_general_ci	N	修改日期
created	varchar	128	N	utf8_general_ci	N	建立日期
modifier	varchar	255	N	utf8_general_ci	N	修改者
creator	varchar	11	N	utf8_general_ci	N	創作人
version	int	11	N	utf8_general_ci	N	修改版本
tags	varchar	255	N	utf8_general_ci	N	分類標籤

表 2-12 資料庫表格 tiddly_Wiki_entry_version

表格名稱	tiddly_Wiki_entry_version	說明	ccTiddlyWiki 之資料表，記錄 Wiki 內所有資訊			
欄位名稱	型態	大小	主鍵	校對	NULL	說明
id	Int	11	Y	utf8_general_ci	N	auto_increment
title	vchar	255	N	utf8_general_ci	N	Wiki 標題
body	text		N	utf8_general_ci	N	標題內文描述
modified	vchar	128	N	utf8_general_ci	N	修改日期
created	vchar	128	N	utf8_general_ci	N	建立日期
modifier	vchar	255	N	utf8_general_ci	N	修改者
creator	vchar	11	N	utf8_general_ci	N	創作人
version	int	11	N	utf8_general_ci	N	修改版本
tags	vchar	255	N	utf8_general_ci	N	分類標籤

2.7 Wiki 在圖書館的應用

Wiki 可應用於社群指引 (Community Guide)、主題資源指引 (Subject Guide)、圖書館內部應用(Wiki Internal Library Wiki)、內容管理系統 (Content Management System)、討論會的 Wiki (Conference Wikis)、知識庫平台 (Knowledge Base)、研討會計劃的 Wiki (Conference planning)、合作共同編輯文件 (Collaborative editing of documents)、會議之前的議程及相關討論 Meetings (posting agendas, discussion prior to meeting, etc.)、專案平台 (Project space)、線上課程 (In place of structured courseware)、呈現的媒介 (Presentation medium)、做為組織思考的線上記事本 (Web-based notebook to organize your thoughts) ...等[41]，只要任何想像的到的主題應用，符合「線上」「共同參與合作」特質，幾乎都可以立即的套用到 Wiki 平台。

愈來愈多的圖書館運用 Wiki 來管理他們的內部資訊、主題資源指引、內容管理、會議之前的議程及其相關討論、線上課程、圖書館利用指引等等。例如英國的哥倫比亞大學 (British Columbia)，利用 Wiki 來管理他們的課程 (Course managent)、會議計劃、共筆撰寫科技文章。有些參考館員則是應用它來當成線上編寫工具，即時的整理業務相

關資訊，進而形成知識庫平台，分享給同仁或是當作教育訓練的教材。例如 Oregon State University 所提供的虛擬參考服務 (<http://oregonstate.edu/~reaset/RDM/>)，利用 Wiki 整合到它們的 Reference Desk Manager (RDM)系統，讓全校師生可以利用此平台，部份則導入到應用於支援教學活動內[42, 43]。有些則應用於圖書館利用指引 (Library instruction Wiki)，例如 Butler University (http://www.seedWiki.com/Wiki/butler_Wikiref/)、The MIT Engineering and Science Libraries B-Team (<http://www.seedWiki.com/Wiki/b-team/>) 所建置的 Wiki 服務[44]，Farkas 應用 Wiki 平台，進而建立 Library Success : A Best Practices Wiki (<http://lib.success.org>)，藉以讓分散於世界各個角落的有心人都可以自由的分享以其實務相關經驗，圖書館應如何引進 Wiki 的技術，藉以輔助平日所面臨到的業務，或許可以參考現行已在使用 Wiki 服務的圖書館，表 2-13 即為目前現行各個圖書館的應用整理 [45]。

表 2-13 Wiki 在圖書館的應用

項次	項目	說明
1	University of Calgary Wiki / Canadian Library Association	<p>http://wiki.ucalgary.ca/ Calgar 大學委託教學中心 (http://tlc.ucalgary.ca/) 所建置，該 Wiki 並非圖書館所專用，主要是提供 Calgar 社群，共同教學與學習的合作，及內容管理的空間。</p> <p>http://Wiki.ucalgary.ca/page/CLA 仿效 M. Farkas' ALA Wiki 的作法，建置 CLA Wiki Page，讓參與會議的學者，能更快速地獲取該次會議的相關資料。該平台採用的 Wiki 引擎為 pmWiki。</p>
2	Koha Project Wiki	<p>http://Wiki.koha.org/doku.php Koha Project Wiki 提供架設 Koha 圖書館系統安裝文件、FAQ、Bug 回報等等相關文件，該 Wiki 引擎採用 WikkiTikkiTavi。國內輔仁大學圖書資訊學系毛慶禎教授亦積極推廣 Koha 軟體，也成立了 Koha Wikispaces(http://koha.wikispaces.com/)分享相關資訊，平台內有許多中譯文件可供參考。</p>
3	LIANA/ITSIG Wiki	<p>http://Wiki.lianza.org.nz/ 起源於 2003 年 10 月份，成立的最主要目的為給紐西蘭圖書館員和資訊管理人員機會探索 Wiki 的用途，主要是為了讓紐西蘭圖書館員和資訊人員，可以進行線上合作發展、描述這些以網路為基礎相關資源。平台內提供資訊科技的應用及管理的相關資訊、ITSIG 與其相關活動報導的資訊、SIG 研究註冊相關資料 (a research register for the Research SIG of LIANZA)...等</p>

項次	項目	說明
		相關訊息可供參考。該平台採用的 Wiki 引擎為 pmWiki。
4	Library Success : A Best Practice Wiki	<p>成立於 2005 年 7 月，由 M. Farkas 所創造及維護，Farkas 主要是觀察到有許多的 Blog 所分享的專業資訊過於分散，使用者僅能透過 RSS 的訂閱方式來取得，或是必須逐個到 Bog 獲取訊息。這些資源過於分散，也無經過良好的組織。故成立 Library Success，希望能將相關資訊集中，讓大家可以在這個園地發表，共同參與撰寫以 Wiki 來做相關服務的經驗集中地。</p> <p>平台內，可以獲得那些已經使用 Wiki 的用戶，分享他們的經驗。這些寶貴的資訊，可以讓要剛要踏入這塊領域的圖書館迅速的獲取有用訊息，減少不必要的遠路，目前的分類，區分成 12 個大類。使用者可以直接對其所需要的部份，直接進而閱讀。例如，它裡面有一份 Wiki page (How Libraries Can Use) http://www.libsuccess.org/index.php?title=How_Libraries_Can_Use，就介紹目前圖書館可以怎麼樣使用 Wiki 來輔助圖書館的相關業務。該平台採用的 Wiki 引擎為 MediaWiki。</p>
5	LISWiki (Library and Information Science Wiki)	<p>http://www.lisWiki.com/</p> <p>主要目標在建立任何人都可編寫的百科全書，和維基百科 (Wikipedia)最大的不同點在於，可以找到更多與圖書館學相關資料。該平台採用 Wiki 引擎為 MediaWiki。</p>
6	Library Instruction Wiki[46]	<p>http://instructionWiki.org</p> <p>Oregon Library 所建置的利用指導 Wiki (instruction Wiki)，主要用來協助圖書館的讀者，教導如何使用 Wiki 來呈現的資料，尋找課程大綱、提供建議閱讀的清單...等用途，也利用 Wiki 平台，來實踐集體的研究討論。該平台採用 Wiki 引擎為 MediaWiki。</p>
7	Library 2.0 Wiki	<p>http://Wiki.librar2.net/</p> <p>主要用來讓各界圖書館討論 Library 2.0，專注於收錄 Library 2.0 相關議題。網站內提供 Library2.0 相關計劃，也探討 Web 2.0 相關技術的指導方針與建議。該平台採用 Wiki 引擎為 MediaWiki。</p>
8	SJCPL's Subject Guides	<p>http://www.libraryforlife.org/subjectguides/</p> <p>美國印第安納州的公共圖書館 (St. Joseph County Public Library)所成立的主題資源指引，該 Wiki 平台的使用僅開放給館員有權限去編輯、新增。截至 2007 年 1 月 16 日為止已建置 17 種分類，34 個主題可供參考。網站內的資料.搜尋方式可依館方所分類好的條目進行搜尋，或是直接以站內所提供的搜尋引擎進行搜尋。</p>
9	Ohio University	http://www.library.ohiou.edu/subjects/bizWiki/

項次	項目	說明
	Libraries BizWiki	為美國俄亥俄州中大學圖書館館員所維護，主要收錄參考書籍、資料庫、網站、及其它研究型指引網站等相關商業資訊，建置的目的是希望幫助商業研究者可以快速地針對進行中的專案或研究找到最佳的資源。系統內所提供指引資源，幾乎可從 ohio 校園內存取到全文。平台內較為特殊的是將 Web 2.0 Meshup 機制導入系統內，將 IM 整合至 Wiki 平台內，使用者可透過 my widget 將軟體安裝在本身所使用的電腦，當使用上發生問題時，就可直接詢問館員。
10	The MIT Engineering and Science Libraries B-Team	http://www.seedWiki.com/Wiki/b-team 為 MIT 所建置，收錄生物學相關主題，其目的在於協助，支援該校的研究與教學單位。採用的 Wiki 引擎為商業網站所提供之軟體 (http://www.seedWiki.com/)。
11	Unofficial Wiki for the 2005 ALA Annual Conference in Chicago!	http://meredith.wolfwater.com/Wiki/ 主要收錄芝加哥及美國圖書館協會研討會相關資料。網站由 M. Farkas 所建置，主要目的是讓讓參加 ALA 2005 Annual Conference in Chicago 的學者，可對該年會有更多的參考資訊可供參考。例如芝加哥餐館的相關資訊、會議的議程、官方及非官方的事件等參考訊息。該平台採用 Wiki 引擎為 MediaWiki。

由現行各個圖書館的應用整理表中，如 The MIT Engineering and Science Libraries B-Team(<http://www.seedWiki.com/Wiki/b-team/>)、Butler University Library(http://www.seedWiki.com/Wiki/butler_Wikiref)，可得知部份圖書館並不自行建置 Wiki 平台，而直接引入商業性網站所提供的免費 Wiki 服務，國內圖書館或許可考慮採此機制的方式導入至部份業務內，例如將主題式資源指引、館員之工作手冊、教師著作目錄...等服務，皆可考慮直接引用現成的商業性網站所提供的 Wiki 服務。採用此機制的好處在於圖書館可減少部份投入人力、維護主機的成本、工作流程客製化可委託商業性網站公司代為處理等。提供另一種選擇方案，不僅建置成本低(主機維護費用)，也可很快速地完成推廣與運作，而圖書館僅需透過線上申請就可以很快速的把 Wiki 服務引入單位內直接應用，目前所提供免費 Wiki 主機服務者(Wiki Hosting)，整理如表 2-14。

表 2-14 免費 Wiki 主機服務者清單

序號	項目	網址	wiki 引擎	語言	備註
16 1	Wikia backpackit	http://www.wikia.com/wiki/WikiBackpackit.com/	MediaWiki	英文 英文	付費版本擁有更多的頁面及上傳檔案數量；目前已超過 20 萬人使用
17 2	Wikidot EditMe	http://www.wikidot.com/ http://www.editme.com/	custom wiki engine written in Java	英文 英文	需付費使用,最低費用\$4.95 美元,可存放 50MB 的容量
18 3	Google Docs & Spreadsheets	http://www.wikispaces.com/ http://docs.google.com/writeboard	Custom	英文 英文	公開的頁面免費,私人使用付費
19	Writeboard	http://www.writeboard.com/		英文	
4	google wiki	http://www.jostspot.com/ http://www.xwiki.com/	Custom	英文	google 收購 Jostspot
20 5	XWiki Groupswiki	http://www.xwiki.org/xwiki/bin/view/Main/Welcome	Custom	英文 英文	少於 100 個頁面及影像檔者免費
21	Jostspot	http://www.jostspot.com/about-jostspot.php	Custom	英文	免費版本限制使用頁面
22 7	Zoho wiki openserving	http://wiki.zoho.com/ http://www.openserving.com/	MediaWiki	英文 英文	
23 8	atwiki(@wiki) pbwiki	http://atwiki.com/ http://pbwiki.com/	Custom Custom	中文 英文	頁面無限制,檔案可存放 10M
9	pttwiki	http://wiki.pttc.com/	Tavi	英文	
10	Schtuff	http://www.schtuff.com/	Custom	英文	3 your.schtuff.com spaces!;Lots of room (200 MB per space)
11	seed Wiki	http://www.seedwiki.com/wiki/seed_wiki/	Custom	英文	cross-platform WYSIWYG editing;no AD
12	socialtext Workspace	http://www.socialtext.com/	SocialText Open	英文	30 天內免費使用
13	tiddlyspot	http://tiddlyspot.com/	tiddlywiki and ..	英文	已使用 tiddlywiki 的使用者,可透過簡單的更改 tiddler,即可把內容上傳到該空間
14	uwiki	http://www.uwiki.com/	DokuWiki	中文	First free wiki farm for Chinese
15	wetpaint	http://www.wetpaint.com/	Custom Java-based engine	英文	申請同時就需決定分類,讓使用者能方便找到相關資源;所產出的文件均採 CC,GNU 或其它授權方式

2.8 Wiki 應用於主題式資源指引網站之探討

圖書館為什麼要採用維基來管理主題式資源指引網站的系統平台呢？圖書館將其導入應用後，又將面臨那些困境或助益呢？先來思考下列可能發生的幾種應用的狀況[47]：

要如何透過一個簡單的機制，可以讓圖書館聯盟的使用者，可以透過統一的介面來一同來創造、編寫維護同一個網站。

1. 專業聯盟在尋找一個簡單的機制，可以讓聯盟參與者可以有一個方便、快速、有效率的方式來更新用戶們的網頁資源。
2. 圖書館經授權的使用者，不需要使用特定的軟體，例如 Dreamweaver 或 Ftp Client Software，就可以很快速、簡單的更新的內部網路資料。
3. 專案在運行，而相關成員分散於世界各個角落，要如何提供一個窗口，讓這些分散的參與者，可以線上即時的分享的經驗給相關的使用者。
4. 會議計劃委員會可能需要一個 Web-based 的工具，可以讓他們即時觀看及修改過往的相關文件記錄，同時也能呈現相關的修改歷程可供參考。
5. 跨組織工作團隊要如何一同合作出版他們的工作經驗、報告等。
6. 如何讓身處於在世界不同的角落的作者，一同地參與會議文獻寫作工作，同時也能觀看到彼此的工作進度、相關內容。而不是只是單方面的寫自己的文章，再透過 Email 傳遞方式來進行溝通。

以上所提到的數種狀況，可以用 Wiki 引擎很快的予以實作。傳統上，當然也可以選擇內容管理系統 (Content Management System,CMS)或 Blog 軟體，來解決以上所面臨的問題。但 CMS 顯少有免費的版本可供使用，且一般也必須透過需要適當的教育訓練，流程客製化等過程才能達到單位的需要。而不管是使用 CMS 或 Blog 所產生和累積的內容都是難以修改，很難產生關連性。最後通常是透過檢索的方式，將一

堆檢索出的結果雜亂的提交給使用者，該使用者自行去篩選所需的資訊。

Wiki、Blog、CMS 與傳統網頁之比較：表 2-15 中比較了傳統網頁 (Web page)、維基引擎 (Wiki Engine) 及 CMS 的不同。使用 Wiki 系統來管理及呈現網站的第一個好處就是把內容的更新程序給簡化了，只要幾個步驟就可以完成所有的更新程序。第二，使用者所用的工具，僅需要瀏覽器，就可以完成所有的修改程序。使用者幾乎不需要改變本身的操作習性。也不用額外熟悉特定編輯 (DreamWeaver) 或更新工具 (FTP Client Software)。使用者只需了解要如何使用網路稍具上網的一些基本技巧。幾乎任何使用者，都可以很快速地使用 Wiki 來撰寫、編輯文件。使用者僅需要專注他們所要產出內容，不再需要額外的學習。由一些實例中，可以看到部份圖書館已採用 Wiki 引擎來架設圖書館的入口網站，例如 Jean and Alexander Heard Library (<http://Wiki.library.vanderbilt.edu/>。上網日期 96/01/23)。Social Software 中，最熱門的軟體還有 Blog 這個選擇。為什麼不採用 Blog 呢？採用 Blog 軟體來實作入口網站的也大有人在。例如吳鳳技術學院圖書館 (<http://www.lib.wfc.edu.tw/>。上網日期 96/01/23)。使用 Wiki 系統與 Blog 系統的一些差異性如下表 2-15。至於要使用者要採用那種系統，來當做主題式資源入口網站的系統平台。研究者認為這完全要視使用者 (開發者) 的目的及方便性來做評斷。CMS 或 Blog 皆可以做為其基礎平台。但不可否認的。Wiki Engine 更適合多對多 (Many-To-Many Communication) 的服務型態。而以 Blog 而言，它仍是屬於一對多 (One-To-Many) 的服務架構。Blog 的內容的分類，主要是「時間」排序為脈絡。而 Wiki 則是以「主題」為脈絡。Blog 主要的開發目的是作為個人發表其想法與意見，再藉由所發表的文章引起讀者討論及迴響。而 Wiki 則是以多人共同編輯建立相關主題為目的所開發的工具。當然它更適於同一主題的共同線上共同編輯的應用。當然使用 Blog 或 CMS 平台不需要特 WYSIWYG Editor (如 Dreamweaver or Frontpage) 工具協助編輯。但它們均需要透過特定的管理介面來管理其內容。而 Wiki 它只要有權限的使用者，就可以自由的新增及修改內容。當然頁面內的資訊，有過時或錯誤的現象時，Wiki 的更新較 CMS 或 Blog 更為方便，快速地獲得更正。以主題式資源指引應用而言，Wiki Engine 則更適合此一方面的應用。


表 2-15 Wiki 引擎、CMS 與 Web page 之工作流程比較表

平台 步驟	Wiki 引擎	CMS	Web Page
1	直接編輯修改頁面	登入系統，輸入帳號	使用特定的軟體來編輯。(dreamweaver、Frontpage,etc.)
2	編輯功能	搜尋要修改之項目	點選你的本地端的網頁檔案
3	編寫、修改的內容	編寫、修改的內容	打開檔案
4	完成更新，直接觀看修改結果	完成更新	編寫你要的內容
5		打開瀏覽器	儲存
6		觀看你上傳的結果	上傳檔案到你的網頁主機
7			打開你的瀏覽器
8			觀看你上傳的結果

表 2-16 Wiki 與 Blog 比較表

平台 項目	BLOG	WIKI
目的	<ol style="list-style-type: none"> 個人發表其想法與意見，藉由所發表的文章引起讀者回應討論。 使用者瀏覽其感興趣之文章。 	<ol style="list-style-type: none"> 多人共同編輯建立相關主題。其內容具有高度完整性。 一般使用者傾向「查詢」資料。
功能	<ol style="list-style-type: none"> 一般功能皆有。 User 無法刪除他人文章，僅能發表迴響。 	<ol style="list-style-type: none"> 功能較為完整，站內資源連結便利。 User 可刪除他人發表內容，但可及時回復內容。
脈絡	以「時間」排序為脈絡	以「主題」為脈絡
分類	僅有一層分類	可層層分類
管理	簡易，多有特定的使用者介面可供使用	頁面繁多，管理不易；需額外了解 Wiki Syntax 的使用方法
介面親和度	<ol style="list-style-type: none"> 操作方式較直覺、簡便 易閱讀 美工介面變化多 	<ol style="list-style-type: none"> 操作較為複雜 內容排版方式較不易閱讀 美工介面變化少，不易修改
推廣參與編輯	易學，一般大眾容易接受	需學會相關語法，年輕族群 / 資訊背景人員較接受

參與社群之選擇：社群(Community) 在傳統上，指的是身處於相同區域有著共同利害關係的群體，社群可以增進彼此溝通狀況、強化信仰系統、提高教育品質、對政治產生影響力，甚致能形成一個政府。隨著時代的進步，網際網路打破了傳統地理距離的限制，只要透過網際網路，就能立即與世界任何一處使用電腦與網路的使用者進行溝通，傳統的社群逐漸轉變成另一種虛擬社群(Virtual Community)，而社群模式(Community Model) 是一種發展線上社群的方法，讓一些個體或團體被受鼓勵，參與在相同目的而設計的持續互動的作法[48, 49]。採用 Wiki 為運作方式的服務，不可以僅思索其優點，尚考慮其它問題，例如參與對象，以維基百科的例子來觀察，創始人 Jimmy 一開始是希望將維基百科設計成一個社群，讓眾人都可參與，剛開始也曾被人說不可行，但 Jimmy 仍貫徹自己的想法。他當時想到是以兩種模式去切入：一種是 Emergent Model，雖然也提供眾人發表內容，但因為考慮到文章的品質與正確性，最後只有少數菁英能上來參與編修內容。最後定會因人數少，流量只停留在某種程度，不會再成長。另一種模式則是 Community model：此模式將編輯和修改權力完全下放給參與者，因為人數和資格不受限制，所以參與人數和文章數量會更多更快，後來 Jimmy 決定採用 Community model，Wikipedia 的蓬勃發展也證明了他的選擇正確。



[36]

Wiki Syntax：Wiki 對於非科技使用者是確實是十分簡單使用的。研究報告中，讓小朋友們(15Grade)實際的體驗 Wiki，讓測試者使用 Wiki 引擎來共同創作線上故事。實驗數據顯示。測試者僅需要短短的十五分鐘就能很順利的使用系統，共同創作，成功的完成複雜的網頁故事 (web-based stories)，該實驗所選用的 Wiki 引擎為 Lizzy Wiki Engine[50]。報告中指出 如表 2-17 及圖 2-14，測試者第一次使用 Wiki 時，最讓人最疑惑的就是要如何為文件創造超文件間的連結、如何管理這些超連結、要如何做好各個文件間的超連結文件的編寫...等，。要如何的把這超文件文本互相給串聯起來，形成完整的故事情節。

表 2-17 使用 Wiki 最常出現之問題統計表

	Frequency							
	Total		Catastrophe		Impasse		Annoyance	
Hypertext	138	62%	72	86%	42	44%	24	53%
Link creation and management	110	49%	66	79%	33	35%	11	24%
Hypertext authoring	32	14%	6	7%	7	7%	19	42%
Image uploading	26	12%	1	1%	25	26%	0	0%
Creating/Editing pages	24	11%	2	2%	6	6%	16	36%
Basic computer skills	19	8%	0	0%	14	15%	5	11%
Collaboration	15	7%	5	6%	4	4%	6	13%
Navigation	6	3%	2	2%	4	4%	0	0%
Global site view	4	2%	0	0%	3	3%	1	2%
Human problems	4	2%	4	5%	0	0%	0	0%
Total	224		84		95		45	

資料來源：[50]

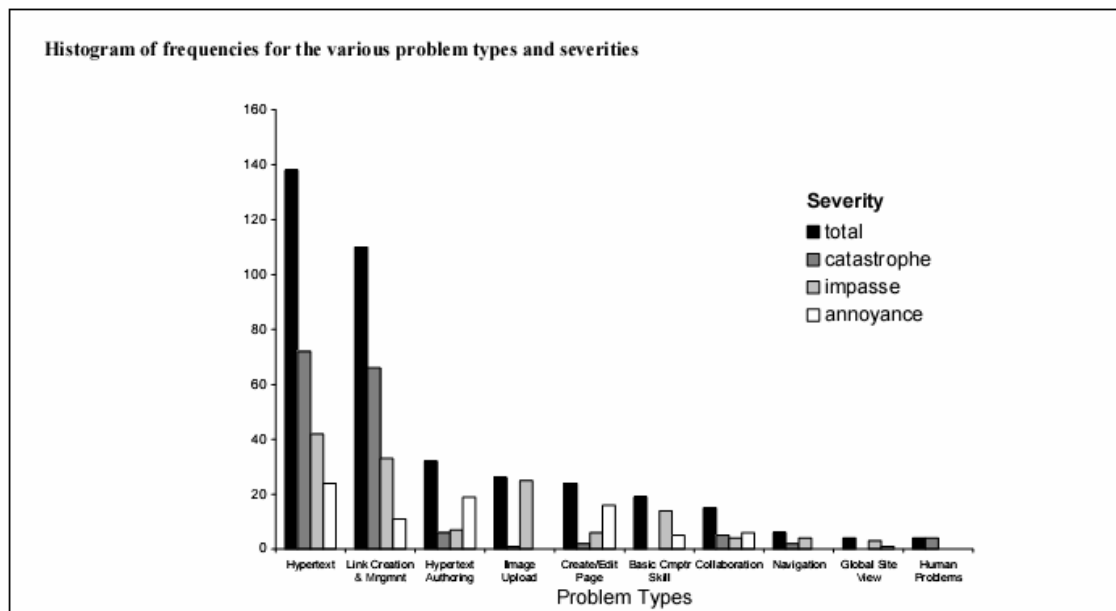


圖 2-14 使用 Wiki 最常出現之問題長條圖

資料來源：[50]

若是以系統面的角度來觀看 Wiki Syntax，Wiki 引擎最主要的用途在於提供基礎平台，讓使用者透過簡單的語法，直接修改線上網頁內容。Script 則是透過維基引擎，把字串存成文字檔或寫入資料庫內。當使用者要從瀏覽器閱覽時，這些腳本再從資料庫內讀取出，Wiki 核心則會解譯字串，將之轉換成維基碼 (Wiki code)。最後輸出能讓瀏覽器所能了解的 HTML 格式。而所轉換的 Wiki code 會嵌入到原本的樣式面 (Template)。進而呈現出 Wiki page 到使用者的瀏覽器上，如圖 2-15 所示。

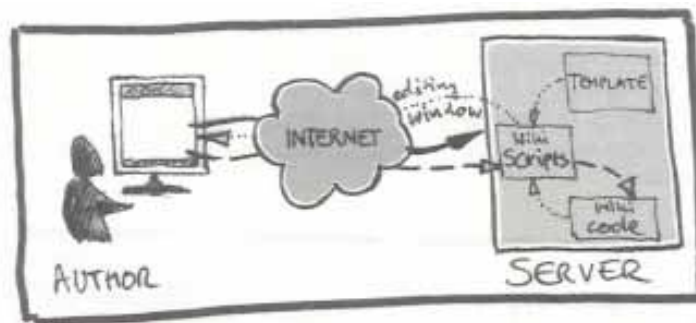


圖 2-15 維基軟體處理維基語法之示意圖 [35]

Wiki 引擎的挑選：表 2-9 開發的程式語言統計表及圖 2-8 Wiki 引擎開發程式語言比例圖中，可以得知 Wiki 引擎的開發的程式語言目前較為廣泛使用的版本計 270 種之多，而所開發的程式語言更達 39 種之數。要如何在眾多版本中選出符合需求的 Wiki 軟體呢？若在一開始不知道該選擇用那種 Wiki 引擎時，建議可以從以下幾個方向開始著手。

(一) WikiMatrix：該網站(www.Wikimatrix.org/)可以直接針對想要使用的 Wiki 引擎直接做列表比較，目前所常見的 Wiki 引擎 幾乎都有涵蓋在內。且系統也可提供一步步詢問的方式，讓使用者能找出最符合需求的 Wiki 引擎。當然適用的 Wiki 引擎會隨著需求的不同而有所不同。藉由 WikiMatrix 所產生的比較表，可直接得知每種不同 Wiki 引擎的特性，例如系統需求、資料的儲存型態方式(資料庫方式或檔案格式)、安全性 /Anti-Spam、發展及支援、是否具特殊功能、 Wiki Syntax 的語法、方便性比較、各方面的統計值(例如最近有無更新版本等)、檔案輸出格式是 XHTML 或 HTML、有無支援 CSS...等比較資訊。

(二) Wiki Choicetree：該網站(<http://c2.com/cgi/Wiki?WikiChoicetree>)，主要的是依據

Wiki 引擎特性來做排序，讓使用者能依具本身的需求直接獲得相關資料。該特性主要分成以資料的儲存方式、SQL 資料庫支援、Unicode、是否使用類維基百科的維基語法 (Wikipedia-style Markup)、使用者權限、Cascading Style Sheets (CSS)、不需經由編輯頁面即可快速改變頁面、在可移動裝置上如隨身碟，有完整功能者、簡單安裝、有無 RSS 功能等等來予以區分。

(三)Google Directory：

http://directory.google.com/Top/Computers/Software/Groupware/Wiki/Wiki_Engines/，主要的分類方式採用程式的開發語言予以排列。簡單的描述該 Wiki 引擎的特性，也提供該引擎連結網址。

(四)維基百科：百科內的 List of Wiki software 條目

(http://en.Wikipedia.org/Wiki/List_of_Wiki_software)，也提供各種 Wiki 引擎清單。主要的分類方式則採用開發的程式語言來分類，讓使用者能依本身所需的系統環境直接挑選合適的軟體。簡單描述其主要的特性。若要獲得比較表，可前往 Comparison of Wiki software 詞條查詢 (http://en.Wikipedia.org/Wiki/Comparison_of_Wiki_software) [45]。

雖然對每個圖書館而言，它不一定是最佳的選擇。以目前的應用觀看，圖書館已經開始利用 Wiki 來實作特定主題的知識庫，內部及內容文件管理系統，當然它也可以用來，讓每個館員交流他們所擁有的資訊。對於一些需要快速且容易在線上協同作業的專案而言，Wiki 確實是一個最佳的選擇。當然 Wiki 在某些特定的專案及特定的目的上，是一件相當好的工具[51]。利用 Wiki 的特性可以發揮社群性軟體的功能，利用它來建置某一特定主題性的知識庫收藏使得主題式資源指引的優點發揮到最高處。

三、 研究設計

網際網路上存在許多有用的資源，包羅萬象、類型繁多、但又隨時變動。愈來愈多的使用者已經把這些網路上的資料，應用在教育、科技、學習以及研究；人們也逐漸地改變使用習慣，尋求資料的方式也改成優先尋找網路上的資源[14]。

面對網路上無窮無盡的資源，圖書館所建置的主題式資源指引，提供了具權威性且經過品質控管的資源，確實可協助讀者有效率地取得高品質的資源。本章將討論現行主題式資源指引網站在實作上窒礙難行之處；而為了解決這些問題，要如何引入當下科技 Wiki 軟體；將 Wiki 引入後又將面臨何種挑戰。

3.1 主題式資源指引網站服務現況分析(以清華大學主題資源指引網站為例)

為了解主題式資源指引網站實際運作的工作流程及其問題所在，研究者對清華大學圖書館進行訪談，訪談的問題及回應整理如表 3-1。由訪談結果可知，圖書館在收錄及建置主題式資源指引網站，所面臨的最大問題仍是在於人力、收錄品質及後續維護。館方所建置好的內容仍需考慮到後續的維護，這是因為所收錄的網路資源，除館藏電子書、電子期刊清單及電子資料庫的變動性不大外，其餘網路資源的變動性較大，例如會面臨連結位置失效、頁面內容的資料早已過時...等問題。

其次是所提供資料的品質，若由館員所主導收錄的學科精緻度，很難與本身就是該學科的專家比擬，因為使用者才能真正了解本身的需求，畢竟術業有專攻。而發布網站的方式採用的是 HTML 撰寫網頁，把資訊呈現給使用者，資訊的提供是單方面的推進 (Push)，這是目前主要的方式，但若是資料有需要更正的時候，館方不一定會主動發覺，此時若有一個互動式的系統，具備共同審查、參與推荐的機制，相信對學科使用者及圖書館是有所助益的。

表 3-1 訪談問題及回應

項次	問題	回應
1	學科指引網站所投入之人力有多少？	專業館員一名，工讀生數名。
2	從資料收集，到成立學科指引網站，共計投入多少時間？	從資料收集到成立入口網站，總共花費一年六個月左右。
3	資料的徵集有無標準流程，主要依據為何？	資料的徵集主要是透過相關系所提供。
4	學科指引網站主要的建置技術為何？	主要是採用 HTML 的編寫方式，將徵集結果呈現給使用者。
5	有無系統支持主題式資源指引網站的運作？	並無系統可供主題式資源指引網站的運作。
6	資料的更新頻率為何？有無窒礙難行之處？	資料的更新頻率並不頻繁，主要的原因為無多餘人力維護，且建置出來的資料精緻性也有待商榷。校內部分系所也自行整理主題資源，圖書館所整理資源的精緻程度很難與其相比，因為系所單位才是真正的使用者，而本身也扮演著該方面的學科專家。

3.2 以 Wiki 實作主題式資源指引之問題

圖書館採用 Wiki 主題式資源指引網站，最大的優勢在於「參與」和「分享」。讀者可直接參與內容的建置，進而分享建置成果。讀者僅須過瀏覽器就可以直接地進行資料的編輯、修改甚至刪除，並可主動新增相關主題資源，發現有錯誤資訊時也可直接更正，網頁的發布不須透過繁複的管理介面或任何編輯網頁軟體的協助。此外，參與社群的對象也可彈性的選擇，例如採用無限制開放或鎖定固定的參與對象，所投入的成本僅有 Wiki 主機的硬體設備與相關維護人力。事實上主題式資源指引網站所面臨最大的困境在於後續的維護，而採用 Wiki 正可解決部份現行所面臨的困擾，由 Wikipedia 等案例中可發現利用 Wiki 線上合作編寫的特性，可以改變現行建置主題式資源指引網站的模式，此外也可以推敲出採此一機制運作方式是能有效節省大量的人力成本，對圖書館而言也許它並不是最佳選擇，但對於需要快速且容易在線上合作共同作業的專案而言，確實是目前僅見的最佳選擇方案[52]。

不可否認的，若以 Wiki 引擎來實作主題式資源指引網站會有諸多問題產生，若從 Wiki 本身的問題點來加以評估，則可參考 M. Farkas 所提出的 Wiki 優缺點，如表 3-2。而將 Wiki 應用於主題式資源指引網站，研究者認為仍有相關的問題產生。結合 M. Farkas 之觀點，綜合整理出表 3-3，分述如下，相關的解決方案留待本章第四節討論。

1. 內容未經組織(Disorganized)：Wiki 系統內的文件由於是共同建置的緣故，任何使用者皆有權可編寫，採此一機制所共同建置出的內容，若未經適當的管控與分類，則產出的內容僅能透過搜尋引擎來尋找，難以接透過分類來進行搜尋。要如何進行共同編輯的文件，有效地進行組織與整理，是一大挑戰。
2. 權限過於開放(Too open)：Wiki 社群的取得方式，通常是透過線上申請頁面進行申請，使用者甚至不需要留下個人基本資訊(如 E-Mail)，不需任何的審核即可完成申請註冊，立即與網站進行互動。而主題式資源指引網站為了監控品質性，參與社群的篩選是有其必要性，要如何在 Wiki 系統內提供有效的控管方式，雖然這與 Wiki 的精神相違，但質與量要如何兼顧與取得均衡點是值得深思的。
3. 被人惡意破壞(Vandalism)：使用 Web 2.0 軟體的前提就是必需信任使用者，但

不可諱言的，Wiki 仍是會發生被人惡意破壞的現象，要如何有效的預防該事件的發生，則可藉由挑選額外的功能來預防，例如 Page History、Page Revisions 等功能，方便追蹤頁面的編輯狀態以防止惡意破壞的現象。

4. 垃圾條目(WikiSpam)：Wikispam 是指在 Wiki 引擎上濫發一些無意義的主題，這些垃圾條目通常都是些廣告資訊，這影響著網站內容品質性。
5. 如何進行挑選 Wiki 引擎：目前而言 Wiki 引擎總計達 270 種之多(截至 2006 年 9 月 27 日止)，要如何挑選出適用的系統呢。
6. 提供更為多元的檢索方式：圖書館館員總希望所建置的資料，可以提供更多的檢索點，讓使用者可依不同條件組合來尋找資料，避免遺珠之憾。但是在 Wiki 引擎中僅能支援全文檢索，要如何將詮釋資料導入到現有的 Wiki 引擎內。
7. 品質控制的問題：要做到品質的控管，Wiki 系統就必須提供管理的機制。將 Wiki 應用於主題式資源指引網站，採共筆機制所產出的內容，所產出的資訊可能是具有爭議性的，導入控管品質的機制，由學科專家與圖書館館員監控，則可改善該問題的發生。
8. 參與社群：掌握 Wiki 是否成功的關鍵因素在於參與社群，而主題式資源指引網站的建置主題確定後，要由哪些使用者共同參與，是透過學科專家以少數人的方式來建置或是擴大到系所助教或甚至到全校師生，甚至全面性的開放，讓有興趣的志願者參與，這延伸出認證的相關問題。
9. 網路上資源無限，要如何提供管道，讓使用者藉由 Wiki 來協助主題式資源指引網站內的資源發掘：要如何提供推薦窗口，讓使用者方便、直覺地主動推薦高品質資源。
10. 更為友善的管理介面：Wiki 是採用 Wiki 語法(Wiki Syntax)的方式來編寫 Wiki 網頁，管理者雖可透過 Wiki 介面直接進行審視與修改；讀者亦可利用 Wiki 介面進行推薦但使用上仍需額外學習 Wiki Syntax。資源的審查與推薦應可提供 Web Form 方式，方便作業。
11. 審核流程的安排：要確保主題式資源指引網站內所提供的內容是具品質性，則管控機制的引入是必要的，應提供符合實際作業工作流程可減輕同仁的工作負

擔，增進工作效率。

12. 審核後的內容，不希望任何人去做更動時應如何處理：由於 Wiki 平台設計理念是開放的，採用 Wiki 所建置的任何內容是隨時可獲得修正的，但是這某部份應用上卻會是衝突的，例如在實作上會面臨某些內容是不希望使用者可做修改的，只需要讓部份欄位(例如網址)可供修改時，此時就需要管制頁面的功能。
13. 統計功能：「大學圖書館應定期進行館藏、讀者服務及技術服務之調查統計，實施績效評估，並據以改進服務品質[26]，Wiki 引擎並無相關功能，應提供相關統計功能，可提供建置人員參考。

表 3-2 Wiki 的優缺點分析表

優點	缺點
簡單使用 *網頁基礎 Web-based *簡單的語法 Simple syntax 每個人都可自由編輯 免費且開放授權碼 彈性且容易擴充功能	擾亂無架構 過於開放 *作者編寫的內容不希望非作者本人修改 易遭惡意修改及垃圾郵件攻擊

(資料來源：http://www.libsuccess.org/index.php?title=Why_Wiki。上網日期：2007/04/17)

表 3-3 以 Wiki 實作主題式資源指引的問題

項次	問題點
1	內容未經組織(Disorganized)
2	權限太過開放(Too open)
3	被人惡意破壞(Vandalism)
4	垃圾條目攻擊(Spam)
5	如何挑選適當的 Wiki 引擎
6	提供更多的檢索點(Metadata issue)
7	品質控制的問題(Quality Control)
8	認證的問題
9	網路上資源無限，要如何利用 Wiki 來協助主題式資源指引網站的資源發掘
10	友善的操作介面

項次	問題點
11	審核工作流程的安排
12	審核後的內容，不希望再去作更動
13	統計功能

3.3 Wiki Engine 挑選

本研究在挑選 Wiki 引擎時主要的考慮因素包含：(一)系統的效能、(二)UI 的設計(美觀及操作性)、(三)功能性、(四)開發語言。研究者透過 WikiMatrix 網站進行挑選，透過比較表(圖 3-1 比較圖)可得知 TiddlyWiki/ccTiddly 與 MediaWiki 的特性，WikiMatrix 雖未收錄 ccTiddly，但 ccTiddly 可說是 TiddlyWiki 的伺服器版本，仍可藉由 TiddlyWiki 得知相關功能，例如系統需求、資料的儲存型態(採資料庫或檔案)、安全性/Anti-Spam、發展及支援、特殊功能、Wiki syntax 的用法、方便性、各方面的統計值(如最近有無更新版本等)、輸出格式(XHTML 或 HTML)、支援 CSS 與否...等相關資訊。

研究者本身較為熟悉 PHP 程式語言，故在選擇時以 PHP 語言所撰寫 Wiki 軟體為主，初步的挑選是選用 MediaWiki 與 ccTiddly 系統。實際架設 MediaWiki 與 ccTiddly 可發現，MediaWiki 系統在剛完成安裝後，直接進行測試，可發現網頁的回應時間長，MediaWiki 官方網站內亦有提供加快的方式，例如額外安裝 eAccelerator(<http://eaccelerator.net/>)或 Squid 伺服器，需額外安裝其它元件；而就 UI 設計而言 MediaWiki 較為單調，採用 Mediawiki 的網站皆具有相同的風格，不似 ccTiddly 可透過 CSS 的修改即可呈現多樣貌風格，就操作性比較使用者很難直覺地使用 Mediawiki(例如新增條目時，使用者必須要在本身的帳號內，在該頁內鍵入[[新條目]]，才能串連到新的條目，開始進行新增條目的動作，初次使用的使用者很難理解這樣子的操作方式)，就使用者介面而言，研究者認為 ccTiddly 設計較佳，使用者可直覺的進行操作；但就系統所提供之附加功能而言，MediaWiki 則較具優勢，功能完善但這也是造成系統效能回應較慢的主因，MediaWiki 具備了討論功能、註冊功能、AntiSpam 功能、檔案上傳功能、頁面歷史...等功能，但就主題式資源指引網站應用而言，其主要目的是仍是指引「正確、高品質」的資源所在，故討論功能、檔案上傳...等功能是不需要的，系統的挑選主要依據上述四個考量點，最後選擇以 ccTiddly 為開發系統之基礎平台。

	MediaWiki	TiddlyWiki/ccTiddly
版本	1.8.2	2.1.2/0.5.5
最後更新日	2006-10-13	2006-10-05
作者	Magnus Manske, Brion Vibber, Lee Daniel Crocker, Tim Starling, Erik Möller, and others.	Jeremy Ruston/CoolCold
開發程式語言	PHP (with some OCaml and C)	JavaScript/PHP and JvaScript
授權方式	GPL	BSD
資料儲存	資料庫(MySQL)	檔案/資料庫(MySQL)
作業系統	UNIX, Windows, MacOS X	均適用
安全性/Anti Spam	Yes	No
HTML Tags	部份支援	全部支援
數學公式	Yes	需採用外掛(Plugin)
頁面歷史	Yes	No/Yes
頁面修訂版	Unlimited	None/Yes
RSS 功能	Yes	Yes
語法範例	內部連結：[[Title]] 外部連結： [http://example.org The title] 粗體：" bold "	內部連結：[[Title]] 外部連結 Link： [[aliased http://www.google.com]] 粗體："bold"

圖 3-1 MediaWiki、TiddlyWiki 與 ccTiddly 比較表

(資料來源：<http://www.Wikimatrix.org/>。上網日期：2006/9/27)

3.4 Wiki 實作主題式資源指引網站改善策略

回顧表 3-3 所提及之以 Wiki 實作主題式資源指引網站的缺點，其中內容未經組織、權限太過開放、被人惡意破壞、垃圾條目攻擊等問題，可藉由挑選適當的 Wiki 引擎加以克服，本研究所採用的 ccTiddly 因為會員註冊並不採開放申請的方式，故權限太過開放及惡意破壞的問題並不存在。以下茲針對其它八點問題加以探討：

1. Spam 問題：Wikispam 是指在 Wiki 引擎上濫發一些無意義的條目，通常此類條目都是些廣告資訊。目前有許多工具都是針對 Email Spam 所開發，對於 Wiki Spam 的處理工具則較為少見，處理 WikiSpam 的方法主要有下列三種，本研究採用限定申請註冊後的使用者及加上內容審核制的方式進行管控。
 - 常規的管控方式：進行人工檢視，發現不當的內容則手動進行刪除並暫時/永久封鎖系統內的問題使用帳號。
 - 限定申請註冊後的使用者：使用者提供完整的個人資訊且經完成申請程序後才能使用 Wiki 系統，例如 DoNews。此種方式一定程度上損害了 Wiki 的開放性。
 - 外掛功能：限制來源 IP，如 Mediawiki 軟體所提供 Spam Blacklist 功能 (www.mediawiki.org/wiki/Help:FAQ#Anti-spam)；或者使用特定的標籤，如 (`<nofollow>`)，用以禁止搜尋引擎對 Wiki 內容的外部鏈接進行分析，主要是處理垃圾鏈接 (Spam Link)，可降低 Spam 效果。
2. 社群模式 (Community Model) 挑選：社群模式挑選是採用開放註冊的方式或是採其它方式獲得解決，此問題對圖書館而言絕不是什麼大問題，因為圖書館本身就擁有自己的使用社群，也就是圖書館的讀者。不管採用那種 Wiki 引擎，都必須修改原有的認證模組，將 Wiki 與圖書館自動化系統的認證模組加以串連，圖 3-2 為更改原有認證方式的流程圖。
3. 品質控制 (Quality Control)：透過共筆方式所建置的資料，其品質是否能為館員所接受？由於 Wiki 的設計理念及運作精神是所有使用者皆可對任意條目執行新增、編修...等動作，而此種方式如果直接應用在圖書館內，它所產出的文件內容，馬上就

面臨挑戰與質疑。該如何引入一個機制，能為其把關？為此，本研究修改了原有的 Wiki 工作流程，引入審核機制，讓學科專家及館員可介入控制條目品質，如圖 3-3，經學科專家或館員經審核編目後的資料，才會呈現至所指定的分類頁面，未經審核通過的條目，則靜待自然的淘汰或再次修改。

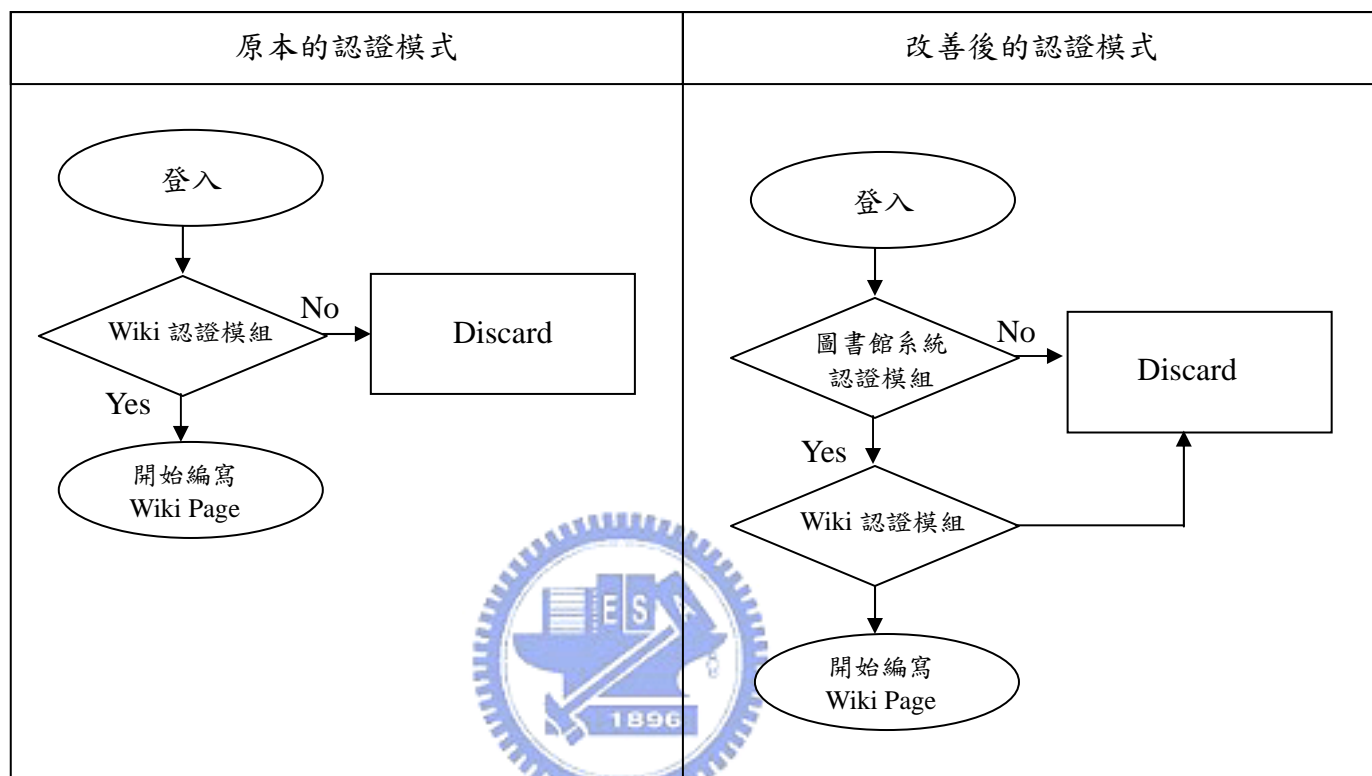


圖 3-2 認證模式流程圖

4. 詮釋資料：系統否能提供更多的檢索方式進行查詢。圖書館所建置好的資源總希望能提供更多的檢索點(Metadata issue)。以 ccTiddly 而言，僅能提供全文檢索，當資料龐大時，則以全文檢索所檢出的資料會雜亂無章。而要提供更多的檢索方式，則需對該筆描述資料提供更多的資訊，必須導入詮釋資料才能改善搜尋引擎。本研究嘗試改變 ccTiddly 原有的資料庫綱要，引入都柏林詮釋資料核心集的 15 個描述欄位。採用都柏林詮釋資料核心集的主要因為，該描述方式簡單、且尤其適用於描述網路資源，各項目欄位易懂易學、語意上的互相轉換、擴張性、欄位具可重覆性、可延伸性、可檢索性等特性。它改善原有資訊組織模式無法適用各種電子資源整理之需要。都柏林詮釋資料核心集目前廣泛運用於網路資源、政府出版品、博物館典藏的詮釋資料格式相關資源的描述。由於它具有以上提及之優勢，故將之引入系統內作為描述資源的基本格式，圖 3-4 為 ccTiddly 資料庫綱要，圖 3-5 則為調整

後之資料庫綱要。

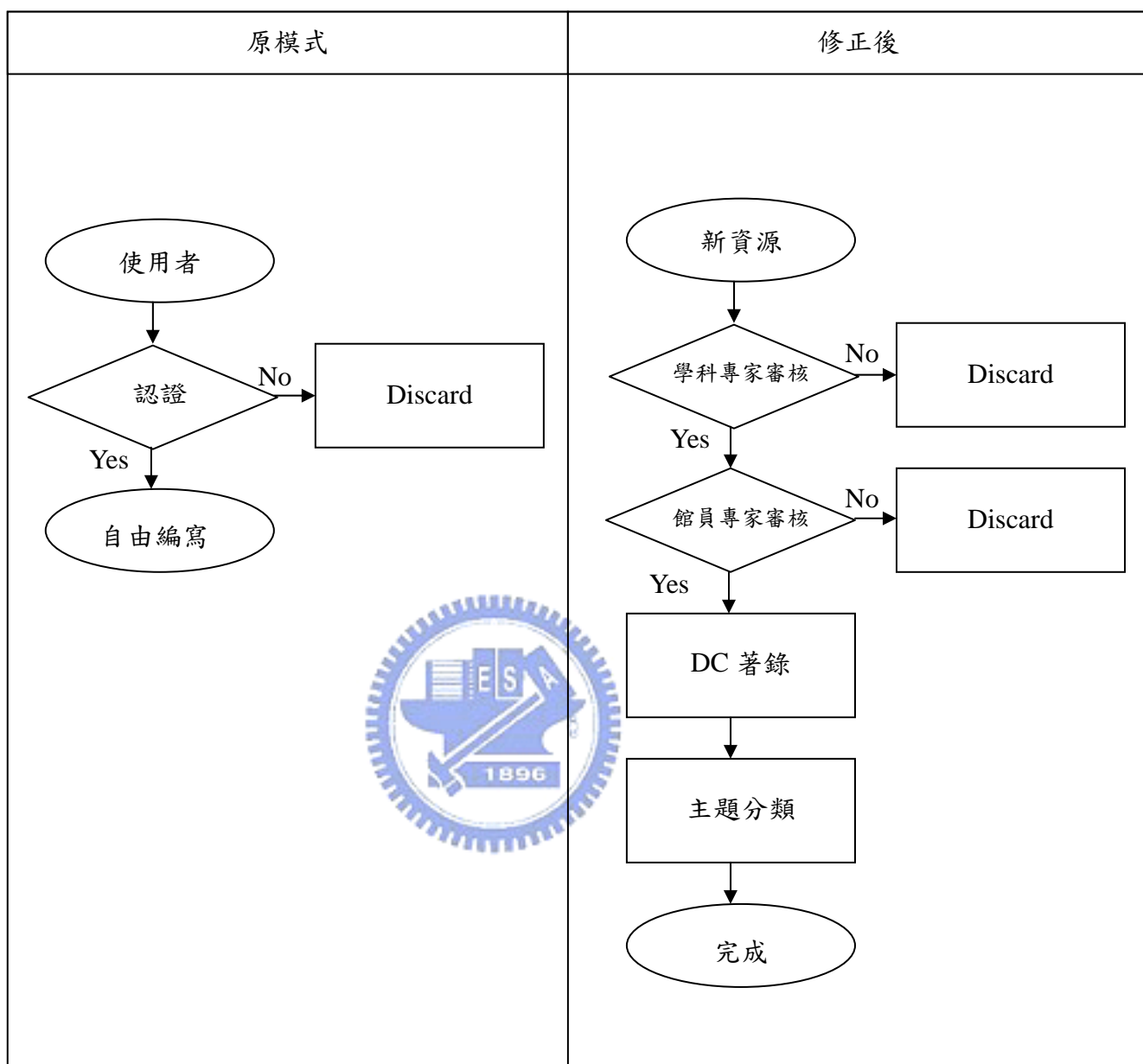


圖 3-3 審核流程圖

伺服器: localhost:3308 ▶ 資料庫: wiki_ming

tiddly wiki entry/ tiddly_wiki_entry_version

欄位	型態	NULL	預設值	用途
id	int(11)	否		系統號
title	varchar(255)	否		名稱
body	text	否		描述內文
modified	varchar(128)	否		修改時間
modifier	varchar(255)	否		修改者
created	varchar(128)	否		創作時間

creator	varchar(255)	否		創作者
version	int(11)	否	0	修改次數
tags	varchar(255)	否		標籤

圖 3-4 ccTiddly 資料庫綱要

資料表	用途	註解
Tiddly_wiki_entry	ccTiddly 資料表	建立：Sep 25.2006, 10:00AM 最後更新：Oct30,2006,03:59PM
Tiddly_wiki_entry_version	ccTiddly 資料表，版本控制用	建立：Sep 25.2006, 10:00AM 最後更新：Oct30,2006,03:40PM
Metadata	記錄詮釋資料	InnoDB free:4096KB 建立：Sep 25.2006, 10:00AM
Ptotect_title	保護頁面功能	InnoDB free:4096KB 建立：Sep 25.2006, 10:00AM
recom	推薦資料	InnoDB free:4096KB 建立：Sep 25.2006, 10:00AM
User	紀錄管理者資訊	InnoDB free：4096KB 建立：Sep 25.2006, 10:00AM

圖 3-5 調整後之資料庫綱要

5. 工作流程安排與資源推薦：網路上的資源無限，僅憑一館之力是無法收錄到全部的高品質學科資源，難免會有遺珠之憾，故需提供一個管道，可讓使用者能主動告知資源所在，使得資源能有效地被發掘且收錄到系統內。以康乃爾大學以 CORC 系統來整理網路資源工作流程(圖 2-2)來觀看現行電子資源收錄，該工作流程十分嚴謹，但以此流程收錄網際網路上的電子資源可說是耗時費力。本研究參考其原有流程更改成圖 3-7 的工作流程。社群可主動協助發掘資源，且將資源透過系統直接推薦，而所推薦資源則由學科專家把關，審慎決定是否接受該資源，館員則專注於資料的分類與編目，系統同時支援 Dublin Core 的擷取，只要網頁本身註錄方式採 IETF RFC2731，系統就能自動擷取，減少館員的輸入動作，完成編目與分類後，系統即完成資訊發布，並將該資源置入於館員所分類好的類目。主題式資源指引網站主要的用途在於指引使用者有用的資源所在，故智慧財產權的爭議性較少，故將該因素排之在外，工作流程重新安排以期減輕館員們的工作負擔，也較符合實際的操作流程(圖 3-6)。
6. 友善的操作介面：Wiki 系統要完成格式化文字、插入圖檔、建立列表、建立外部

連結、建立表格、建立標題、縮排、插入水平線、樣式、WikiWord、使用固定字... 等操作，皆是利用 Wiki syntax 完成，但這對於從未接觸過 Wiki 的使用者，是十分困擾的。故系統的操作介面設計，朝兩方面進行，原 Wiki 功能則讓原本熟悉 Wiki Syntax 專家使用，可提供更為彈性操作，另外設計 WebForm 的介面供管理者與 Wiki Syntax 初次體驗者使用，方便進行管理與資源推薦，藉以減少額外的學習。WebForm 管理介面功能涵蓋修改網站入口標題、副標題、資源的審核...等功能，而資源推薦設計則是明顯地標記出所需要的欄位，例如資源名稱、資源網址、資源描述等欄位供使用者填寫，推薦者一目了然不需額外使用 Wiki Syntax。



Subject Gateway Wiki System

Work flow

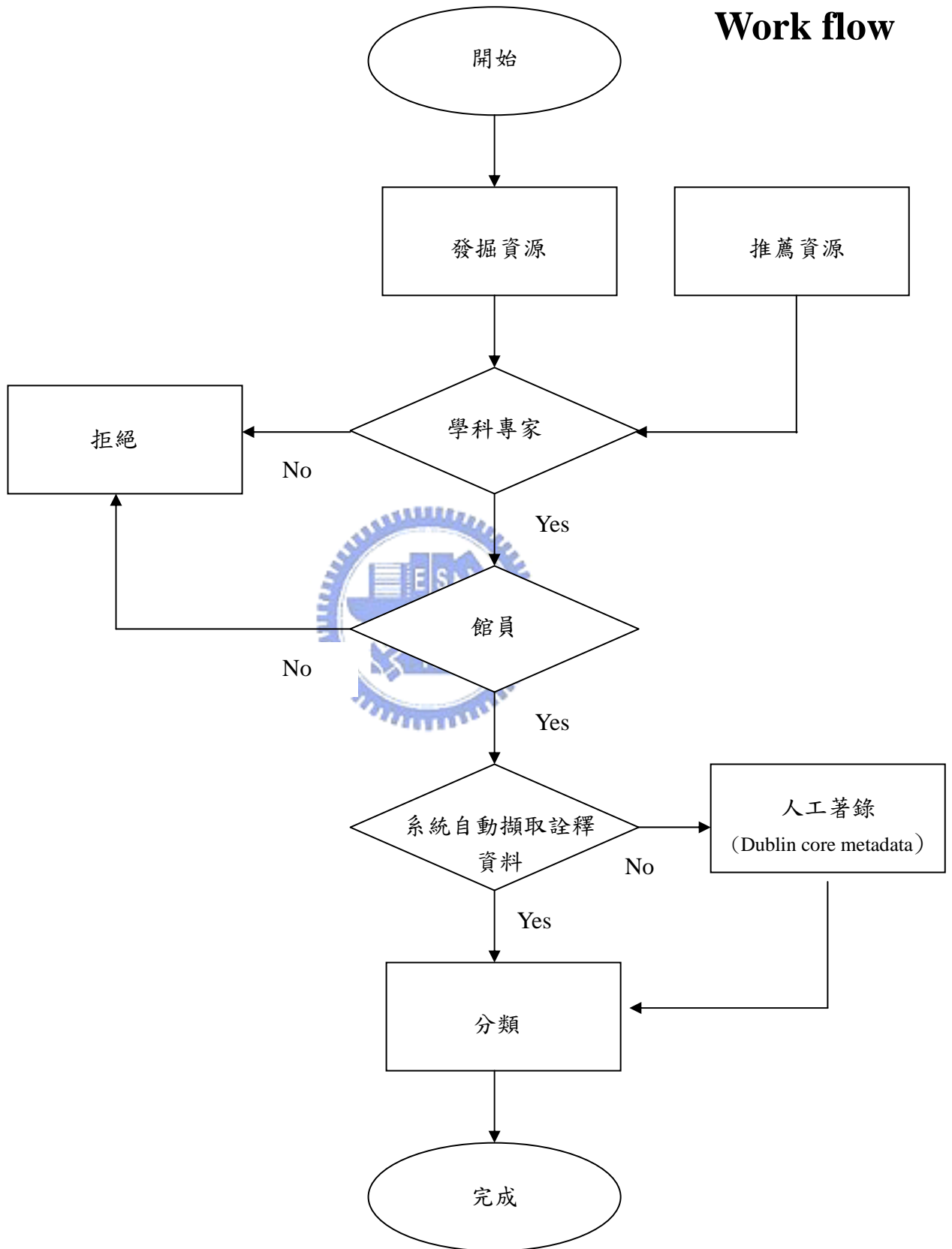


圖 3-6 SGWS 審核工作流程

7. 頁面保護：內容若經過審核後，或是頁面不希望再經變動時，該如何提供保護這些頁面呢？以 Wiki 的機制、精神而言，它是開放的，只要是信任社群，就有權限可以更動頁面資料，但對圖書館而言，要拿 Wiki 來實作主題式資源指引的應用時，網站內的資料，例如入口處的歡迎詞，或是已通過資源審核後的內容，仍不希望已通過認證的讀者可以做修改或更動時該怎麼做；對系統管理人員來說，由於 ccTiddly 的外掛功能可透過 Wiki 頁面完成，只要將讓頁面的標籤(tag)定義為 systemConfig，系統就能立即引入該巨集功能，此功能可說十分方便，但也充滿著危險，此時就需引入頁面保護的機制，將這些不希望改變頁面做限制。
8. 統計功能：「大學圖書館設立及營運基準」第 35 條明確指出「大學圖書館應定期進行館藏、讀者服務及技術服務之調查統計，實施績效評估，並據以改進服務品質 [26]」。系統若無具體的數據可供佐證，服務的績效難以量化，後續的評估、改善亦無法進行，更遑論做出合理、正確的決策。而 Wiki 引擎本身並無考量此一功能，可供使用，在此引入 AWstats 軟體(<http://awstats.sourceforge.net/>)，也額外新增了審核資源統計功能，藉此可分析使用者的使用狀況與系統內實際通過審核的資源總數。



四、系統設計

3.4 節中提出以 Wiki 實作主題式資源指引網站的改善策略。以此為基礎，本章提出系統之設計概念與架構，包括系統設計的目標與規劃、系統架構、系統後置處理等。

4.1 系統設計的目標與規劃

現行主題式資源指引網站主要的實作過程為，透過 Search Engine、Yahoo、其它 SG.. 等方式尋找網際網路內優質資源，經由學科專家或圖書館館員介入，配合著館內既定的相關配套政策進行收錄與過濾多餘資訊，著錄採用一致的標準格式(例如採用 Dublin Core)，去蕪存菁後，再透過靜態網頁或內容管理軟體(CMS)將收錄的資訊發布至使用者，使用者若發覺錯誤資訊時則是透過電子郵件(E-Mail)、WebForm、電話通知的方式連絡網站負責人修改(如圖 4-1)。本研究系統設計的目標在於提供一個良好的平台窗口，改變現行運作模式，資源的收集可透過讀者主動的回報資源所在地，並可提供另一管道徵集許多有用資源，當網站內的資訊描述有誤時，讀者更可直接與以修正。網站內的資料亦有學科專家與圖書館館員的共同把關，維護其品質性，透過系統提供「雙向交流」的可能性，資料的徵集不在是少數人專職，而是提供另一模式，可讓雙方共同的參與、互動、一同建置最為理想的主題式資源指引網站。

本研究開發系統主要採用 Wiki 引擎為主題式資源指引網站基礎平台，取代傳統內容管理系統，應用 Web 2.0 之理念，信賴使用社群一同改善現行主題式資源指引的困境，透過 Wiki 系統可將主題式資源指引網站最為惱人的資料徵集與後續維護等問題，提供另一管道讓使用者共同參與維護(如圖 4-1)。系統功能的設計，引入自動擷取 HTML 文件的詮釋資料功能，自動更新讀者所修改的 URL 網址，於 Wiki 內加入審核機制，審核功能則採用 WebForm 的方式進行，以求減輕館員作負擔，同時為了增豐富入口網站資訊，亦引入混搭(Mashups)相關功能。主題式資源指引網站的內容透過彼此雙方共同合作，相信更能創造出更高品質、更為優質的之主題式資源指引網站。

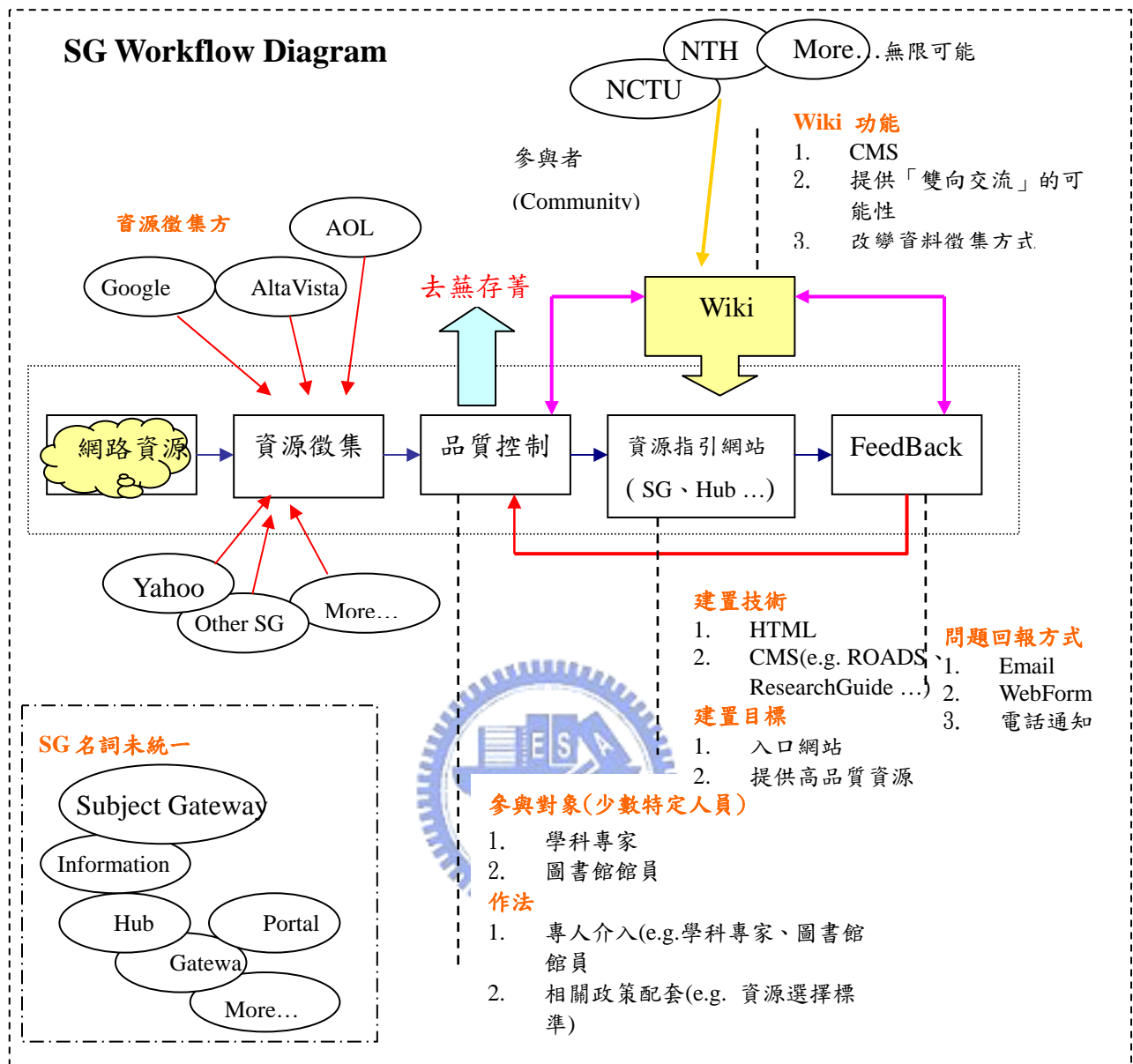


圖 4-1 SG Workflow Diagram

4.2 系統架構

本研究中所設計的 Subject Gateway Wiki System 系統 (以下簡稱 SGWS)，其架構可區分成 Front End、Middleware、Back End 三個層次來表現，如圖 4-2 所示，以下則分別予以說明。

Subject Gateway system Architecture

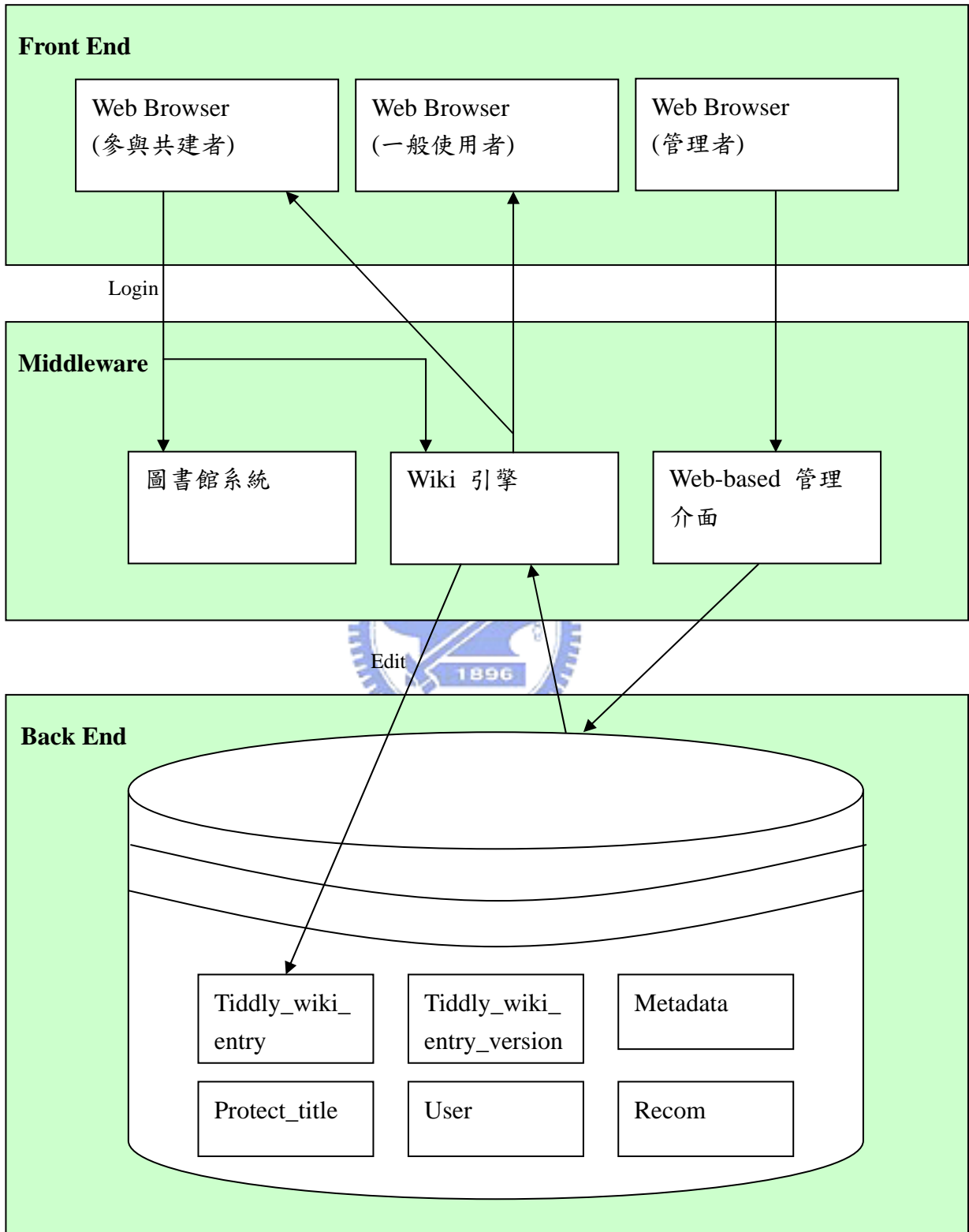


圖 4-2 SGWS 系統架構

4.2.1 外部界面(Front End)

SGWS 系統的使用者主要有三類。第一類使用者僅為單純的使用，乃是利用該主題式資源指引網站獲取資訊。第二類使用者則為進階使用者，參與網路資源的建置與推薦，此類使用者，需經通過認證模組的驗證後才能參與共建。第三類使用者即為管理者，管理者可直接透過管理者介面完成所有的設定，擁有最高權限。

4.2.2 中間層(Middleware)

SGWS 系統中間層的運作可區分為三個模組。第一個模組為圖書館系統，主要是為了認證機制而存在。當然系統的認證可以直接將圖書館系統內的使用者資訊直接複製一份到 SGWS 內的使用者資料庫，但採此一方式會有資料不一致(Inconsistency)的情形發生，故此採用線上即時比對圖書館系統中使用者資訊的作法。第二個模組就是 Wiki 引擎，最主要的用途在於提供基礎平台，轉換使用者所輸入的文字。第三個模組則為管理介面，此介面主要的目的是為減少使用 Wiki Syntax，提供一個簡單 Web Form 使用者介面，應用在審核資源的管理介面、資料保護頁面...等。

4.2.3 後端層(Back End)

SGWS 系統的後端層，也就是資料庫，它主要分成五個資料表。其功用分述如下：

1. Tiddly wiki entry : 主要是存放每筆資料相關資訊的地方，其 Schema 可參考圖 3-4。其欄位主要用途簡述如右：Title 為記錄每筆資源的標題、body 為存放資源描述的地方、modifier、creator 資料表的功用在於紀錄資料的創作者與修改者、modified 與 created 資料表主要是記錄修改時間與創作時間、version 主要用途為版本控制，代表著資料的修改次數、tag 為其分類(標籤)。
2. Tiddly wiki entry version : 此 schema 和 tiddly wiki entry 相同。該資料表主要的功用是為了避免網站內資料遭受惡意毀損，作為版本控制用。
3. Metadata : 該資料表主要目的是記錄每筆資源的都柏林詮釋資料核心集(Dublin Core)描述。
4. Protect_title : 保護特定頁面。

5. User：存放管理者身份帳號的地方。

6. Recom：暫存推薦資料用。

4.3 Wiki 實作主題式資源指引網站改善策略

由 3-2 節中之探討可得知以 Wiki 實作主題式資源指引網站所面臨的困窘，要如何解決所面臨相關問題呢？本研究分別以管理者及使用者的角度探討改善之道。

4.3.1 管理者觀點

Spam 問題、社群模式(Community Model)、品質控制(Quality Control)、工作流程安排、頁面保護與統計功能...等問題是管理主題式資源指引網站管理人員最為關注的，本研究系統所擬定的改善措施，針對改善原有權限管控、資源審核流程、搜尋引擎、頁面保護、網站呈現風格...等相關功能分述如下。

- 權限控管

Wiki 系統所面臨的權限過於開放(Too open)、被人惡意破壞(Vandalism)、垃圾條目(WikiSpam)、參與社群(Community)可透過控制參與社群來獲得改善，而圖書館本身所擁有的讀者(NCTU Community)，正是最佳社群，將圖書館系統認證模組與 ccTiddly 結合，改善原有認證模組，就能迅速建立社群資訊，Wiki Spam 的解決方案可透過限定社群內的成員才有權更動頁面內容，解決 Wiki Spam 的困擾。

ccTiddly 權限控管的方式採用 Javascript 的程式語言撰寫，將使用者資訊值寫入於陣列內(Array)，該認證的模組程式檔置放於 /config/default.php，認證方式主要確認使用者所輸後的值(txtUserName、pasSecretCode 的 Cookie)是否與陣列內的值相等，另外 ccTiddly 考量到安全性的問題，在送出使用者密碼(pasSecretCode)後，會將該值以 MD5 處理後再存入 Client 端的電腦內，部份程式碼如圖 4-3。

```
<?
// database/table
$stiddlyCfg['db']['host'] = "localhost"; //sql host
$stiddlyCfg['db']['login'] = "yourdbaccount"; //login name
```

```

$tidlyCfg['db']['pass'] = "your password";      //login password
$tidlyCfg['db']['name'] = "your db";           //db name

$tidlyCfg['table']['pref'] = "";              //table prefix
$tidlyCfg['table']['name'] = "tidly_Wiki_entry";
$tidlyCfg['table']['backup'] = "tidly_Wiki_entry_version";
// site preferences
$tidlyCfg['pref']['tw_ver'] = "tw_ok_tidlyWiki2.0.10";
$tidlyCfg['pref']['language'] = "en";
$tidlyCfg['pref']['delete'] = 1;
$tidlyCfg['pref']['version'] = 1;
$tidlyCfg['pref']['cookies'] = 0;
$tidlyCfg['pref']['lock_titles'] = $ar_protect
//echo $tidlyCfg['pref']['lock_titles'][0];
$tidlyCfg['pref']['private_tags'] = array();
$tidlyCfg['pref']['free_tags'] = array();
$tidlyCfg['pref']['loginLock_tags'] = array("systemConfig");
//username temp erase
//$tidlyCfg['user'] = array("ming"=>"1234");      //username password
$tidlyCfg['developing']=0;
?>

```

圖 4-3 程式碼 /config/default.php

為導入圖書館系內的有效讀者資訊，則需另外修改原 ccTiddly 系統，作法則有三種方案可供選擇(1)不改寫 ccTiddly 系統，直接將需要的帳號與密碼寫入陣列內 (2)改寫原 ccTiddly 架構，將陣列方式改寫成資料庫方式 (3)將 ccTiddly 與圖書館系統認證結合。方案一最容易實作，但圖書館系統內的使用者的資料，少說也有數萬筆，若採 Array 的方式來實作，不僅影響電腦主機執行效率。亦十分浪費記憶體空間，加上若使用者更改其基本資訊如 Password，更無法即時獲得該資訊。方案二是將使用者的資訊儲存於本機端的資料庫內，這部份需調整 ccTiddly 系統結構，雖可改善浪費記憶體空間的缺點，但

仍會產生資料不一致(Inconsistency)的現象，最佳的作法為第三種方案，將 ccTiddly 與圖書館系統認證結合，使用者的認證資訊會先送到圖書館系統驗證，在將驗證結果送到 ccTiddly 認證模組，本研究採用第三種方案進行認證，更改原認證流程如圖 3-2。

於實作時所遭遇最大的問題為，雙方的變數無法直接進行交換，主要是因為 ccTiddly 權限控管的方式採用 Javascript 的程式語言撰寫，而圖書館認證模組是採用是採用 PHP 語言所撰寫，PHP 語言是屬於 Server-Side Language，而 Javascript 則是屬於 Client-Side Language，簡單的說兩者的程式語言間的變數傳遞是無法直接進行溝通，僅可藉由 Cookie 或 Get 的方式來加以傳遞變數，但使用者的密碼是屬於較為敏感的數據，故不能以明碼來加以傳遞，因此認證的流程則是先將使用者的帳號與密碼先送到圖書館認證系統先與以認證後，在轉到 ccTiddly 內做處理，相關程式碼如附錄一與附錄二。

● 資源審核流程

SG 的可貴之處，在於有學科專家、館員等人的把關，協助使用者尋找專業性資源與過濾大量不相關的網路資源，要如何將該機制導入於 Wiki 系統內，而網站內資源品質若要控制，則管理者必須掌握某部份的控制權，雖然這麼做會使館員工作量增加，但這是無法避免的。以原 Wiki 系統的設計理念，社群內的使用者就有權可自由的進行新增與編輯，但採用此一做法是難以兼顧其品質性。

資源審核的工作流程設計，主要參考以康乃爾大學以 CORC 系統來整理網路資源工作流程(圖 2-2)來觀看現行電子資源收錄，並更改成圖 3-7 之工作流程，使資源的收錄更符合實際工作流程，將資源分類與描述資源詮釋資料的工作給抽離出來，這兩個部份交由學科專家及館員進行掌控。讀者雖可透過 Web Form 的介面或 Wiki 本身頁面新增的功能直接推薦，推薦後的資料雖仍會呈現在網站，但並不會呈現在館員所指定的分類頁面上，另外使用搜尋引擎也不會搜尋出未經審核後的資訊。推薦後的資料會先經學科專家的審視後，在送給相關著錄的館員們進行後續的著錄詮釋資料工作，於編目的同時，系統會自動抓取該筆資源(網頁上所著錄)上的相關詮釋資料(Metatag)，以減輕工作負擔，完成審核後的同時，該筆資源也立即呈現在網站內。

● 詮釋資料及搜尋引擎之改善

ccTiddly 本身雖提供了全文搜尋的功能，但當所建置的主題式資源指引網站日益龐

大，利用全文搜尋所找出的資訊太多，此時仍需花費許多精神篩選資料，全文搜尋的缺點就是無法精準的獲得所需資料。而本研究導入的詮釋資料，就提供了最佳的基礎建設，使資料的搜尋能依不同的面向進行檢索。

詮釋資料採用最為通用的都柏林詮釋資料核心集，更改原 ccTiddly 資料庫綱要如圖 3-5，加入 Metadata 資料表藉以記錄各個資源的詮釋資料。以此為基礎才能提供讓每筆資源可依不同的面向(例如針對作者、不同主題...等)進行檢索。

- 頁面保護

ccTiddly 的開發作者原先即有設想到頁面保護的功能，主要是因為 ccTiddly 系統新增巨集功能可直接透過 Wiki 頁面，將所需功能直接嵌入於系統內，使用者不需藉由修改原始碼，僅需透過 Wiki 頁面的新增即可迅速的導入巨集功能，該巨集 Wiki 頁面即是提供新功能的 Script，當有使用者惡意破壞時，該功能即會失效，透過頁面的新增的方式即可導入新功能，簡單方便但該作法也像是兩面刃，可能會為網站帶來危害，原 ccTiddly 系統對於需要保護的頁面的作法，需修改原始碼，將保護的 Wiki 頁面的 Title 名稱寫於陣列內。本研究之需求是要提供一個簡易 Web-based 的管理介面，可讓管理者很簡易地將特定的頁面予以保護，例如某些經學科專家、館員審核後的特定的 Wiki 頁面，不再希望讓使用者可以自由的修改，就可利用此功能來與以保護。研究者將原陣列方式改寫成以資料庫方式進行，調整原 ccTiddly 資料庫結構，新增 Ptotect_title 資料表，管理者透過 Web-based 介面，簡單的輸入將保護頁面名稱後，即可達到保護頁面之需求。

- 統計功能：

針對統計功能，管理者最想知道的數據是使用者的操作行為與目前網站內有多少筆已經審核完成的資源。使用者的操作行為可以藉由分析網站伺服器內的存取紀錄檔(Log files)，此部份與 Wiki 系統無關，可藉由一些 Open Source 軟體與以解決，不需額外撰寫分析 Log files 軟體；而要統計多少筆資源是經過審核完成後的資料，則必須在 Wiki 資料庫綱要(Database Schema)內有此考量設計，本研究改變原 Schema，新增一欄位旗標值以供判斷，旗標值為"Y"時，代表著該筆資源已經審核通過。

- 網站呈現風格管理

採用 ccTiddly 所建置的網站，所呈現的風格可多樣化，ccTiddly 的樣式表(Style Sheets)架構如圖 4-4。

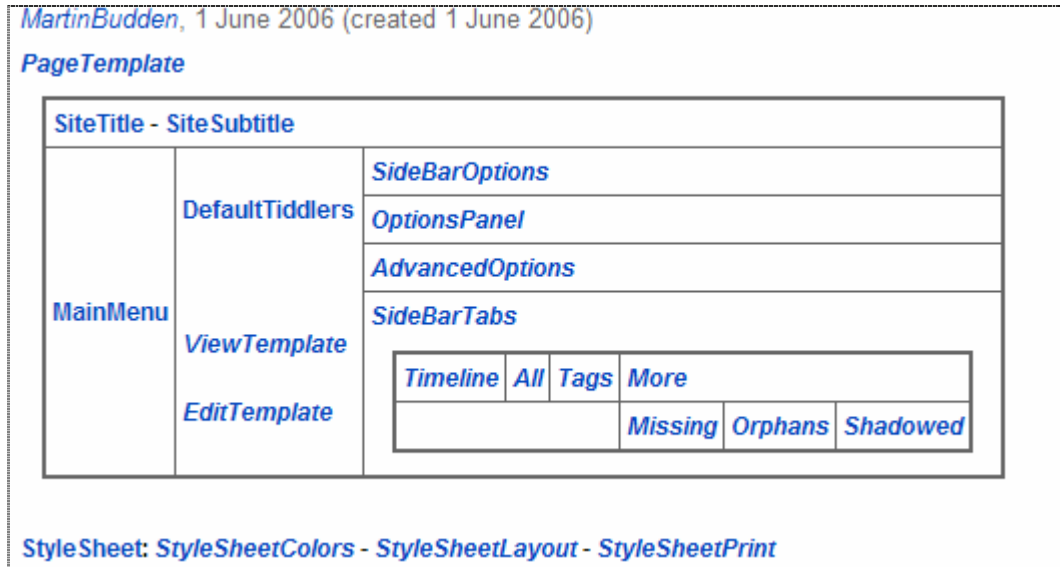


圖 4-4 TiddlyWiki 風格呈現之結構

(資料來源：<http://www.tiddlyWiki.com/#Configuration>)

系統管理者若要更改網站的呈現風格(Style)，僅需修改相關 Tiddler，就能迅速展現不同的樣貌。也就是說透過 PageTemplate、ViewTemplate、EditTemplate 與 StyleSheetLayout 的修改可以快速的進行版面調整，而 StyleSheet、StyleSheetColors、StyleSheetPrint 則與樣式呈現有關。

- # 調整版面
 - * PageTemplate、ViewTemplate、EditTemplate
 - * StyleSheetLayout
- # 調整樣式
 - * StyleSheet
 - * StyleSheetColors、StyleSheetPrint

透過修改相關 Tiddler，就可自訂網站需要呈現的風格(圖 4-5)。當然也可以利用現成的元件，直接將新功能引入系統內，這現成的元件在 ccTiddly 稱之巨集。巨集功能可

讓使用者在不需修改任何原始碼的情況下，直接把所需要的新功能直接應用到 ccTiddly 內。事實上 ccTiddly 可以直接應用 TiddlyWiki 內所提供許多有用的巨集功能，網際網路上也有許多熱心的玩家，提供自家開發的巨集供大家使用，例如像 MonkeyPirateTiddlyWiki 和 TiddlerWiki，在臺灣 BramChen 也不惶多讓地貢獻所製作的巨集供各界使用，本研究網站所採用的 Style Sheets 即為 BramChen 所貢獻的模版。以下即為一些關於 Template、Style sheet 相關的巨集介紹：

1. 快速切換樣式表 (<http://www.tiddlytools.com/>, by Eric Shulman)：

該功能是能夠讓使用者快速切換使用的樣式表及做樣示表的管理，它本身引入的機制可說是 StyleSheet 界的大革命，是無限模組化的可能性，可讓使用者選擇所要呈現的樣貌。

2. 樣式表收集庫 (<http://www.zrenard.com/tiddlyWiki/stylesrepository.php>)：

該網站收錄許多模版可供使用，節省了開發的時間，各式模版都是些令人讚賞的好作品，建議覺得系統預設模版不符合需求時，可至該網站找找適合的模版使用。

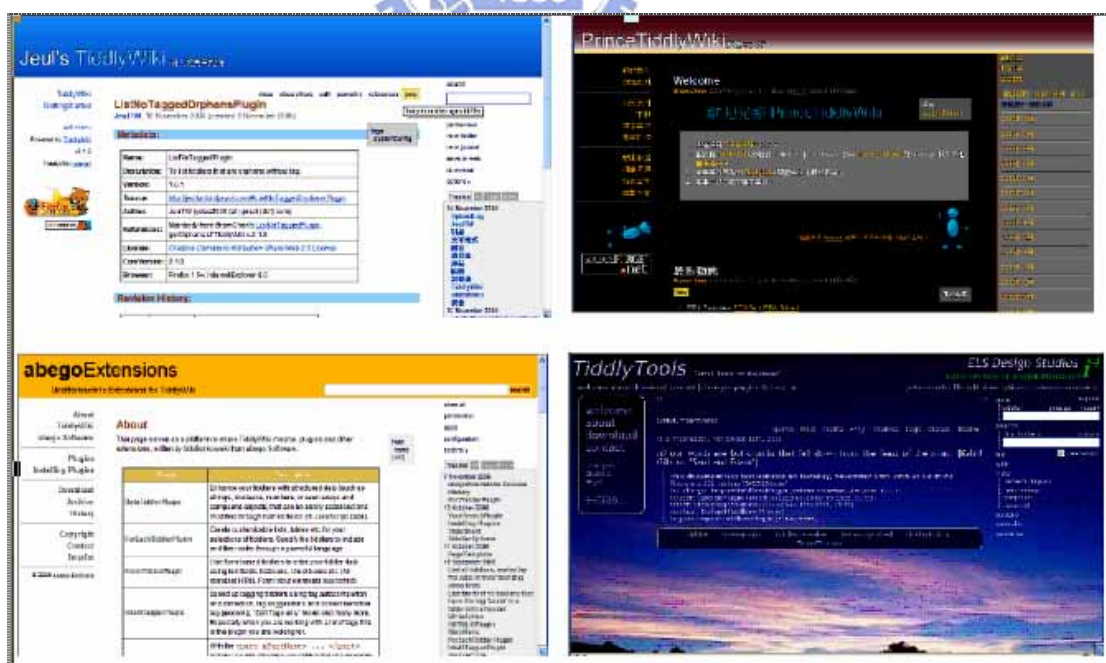


圖 4-5 TiddlyWiki 的多風格呈現

4.3.2 使用者觀點

由使用者觀點來看，網站內所建置的資料提供了諸多有用的主題指引，要如何的搜尋各式資源與參與網站的互動是 SGWS 的重點，以下即針對使用者如何搜尋資料與要如何參與互動進行討論。

- 如何搜尋資料

使用者可藉由「瀏覽」與「搜尋」的方式來進行尋找資料。瀏覽的方式可以直接觀看圖書館所精心製作的資源整理，透過網站左側導覽列，可逐層進行瀏覽，該方式適合初次不知該如何找尋那些資料的使用者。

系統導入詮釋資料的機制後，可提供兩種搜尋方式，第一種為「簡單搜尋」，使用者只需輸入關鍵詞後即可進行檢索；第二種方式則為進階檢索，使用者輸入關鍵詞後，可針對各個特定欄位進行檢索(e.g. 作品名稱)。原 Wiki 系統與改善後之系統最大的差異在於原 Wiki 所提供的搜尋引擎僅可針對資料表內的 body 欄位進行搜尋，而導入詮釋資料概念後，搜尋引擎則能針對各個欄位進行檢索，可讓使用者提供更為精準之搜尋。

- 如何參與共建

系統提供兩種資源推薦方式供使用者選擇，即透過傳統的 WebForm 方式或 Wiki 引擎既有的功能來完成。使用者可透過 WebForm 的推薦的頁面，直接進行推薦，只要輸入所需欄位資料後，按下送出後即完成推薦程序；也可利用 Wiki 系統內的「推薦資源」功能直接使用 Wiki Syntax 新增條目，兩種方式都可達推薦資源的目的，差異在於使用 Wiki Syntax 可讓網頁排版更為彈性與自由。此作法改變傳統的資料徵集方式，傳統 SG 的資料徵集多採單位自行徵集的方式進行(如圖 4-1)，該作法費時費力。現在只要透過網站右上方的導覽列內的「推薦資源」或 Web Form 推薦表單，就可以直接線上參與推薦。

除推薦功能外，當使用者發現網站內的資訊有所錯誤時(例如描述錯誤或連結網址已失效...等)，可直接前往修正，修正方式為在待更正的頁面上，按下「編輯」功能或使用滑鼠在頁面上點擊兩下(Double Click)，就可直接進行修改。簡單、迅速且有效率。

五、系統實作與驗證

在第四章中介紹本研究如何對 Wiki 系統進行修改和擴充，建置了一套適用於主題式資源指引網站的系統 SGWS。本章在於描述 SGWS 的實作結果，首先說明系統的實作環境與開發工具。再來說明系統各細部功能，且針對第四章所提及之各項改善，展示其實作結果。

5.1 系統環境

SGWS 系統的實作開發環境如圖 5-1。整個系統的環境採用 LAMP 架構(Linux、Apache、MySQL、PHP)，作業系統採用 Ubuntu (Unix Like)，網站伺服器為 Apache。後端資料庫則使用目前主流的 MySQL，該版本的資料庫編碼方式 (Encoding) 預設為 UTF-8。而連結前端界面與後端資料庫的程式語言為 PHP。為了資料庫管理的方便性，採用的管理工具為 phpMyAdmin，而不採用 MySQL 本身所提供的工具。以下為各元件的版本相關資訊。

- Ubuntu OS (Linux Kernel 2.6.15-26-686) – 作業系統
- Apache / 2.0.55 (Ubuntu) – 網站伺服器
- PHP / 4.4.2-1build1 – 動態程式語言
- MySQL 4.1.15-Debian_1ubuntu5-log – 關聯式資料庫
- phpMyAdmin - 2.9.0.2 Version – 資料庫管理工具
- ccTiddly Engine (powered by TiddlyWiki ver. 2.0.10 and ccTiddly 0.5.5) – Wiki 引擎

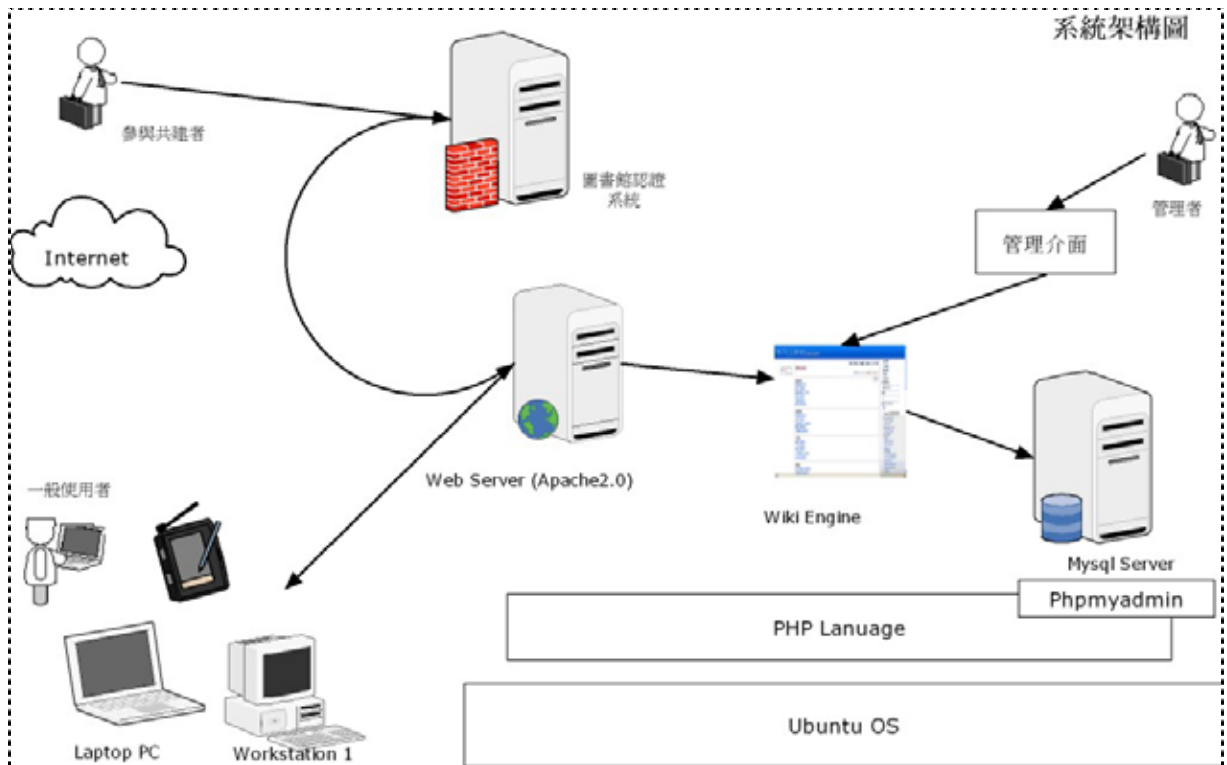


圖 5-1 系統實作環境

使用者區分成三種角色。第一類使用者為「管理者」，管理者透過管理介面可對系統的資料庫進行管理，管理者可藉由此介面進行資源審核、修改網站標題、次標題、增加保護頁面...等工作。第二類使用者為「參與共建者」，該類使用者僅需通過認證就可以透過 Wiki 引擎直接參與推薦資源和修改資源描述。第三類使用者為「單純的使用者」，此類使用者，不參與共建，只是使用網站內資源。

圖書館系統在此所扮演的功能僅為認證。該模組會提供一個 Yes 或 No 的資訊值給 ccTiddly 參考。而中間介面是採 ccTiddly Engine 為整個系統的基本平台，該軟體提供 Wiki 引擎之基礎功能，資料的儲存則是放於 MySQL 資料庫內。

5.2 系統功能

本節在於說明此系統之主要功能。依據 3.2 節所述及之採用 Wiki 實作主題式資源指引網站所需改善的地方加以實作，可設計出如圖 5-2 之系統。

參與共建的使用者群，是通過認證後即可參與推薦資源及修改錯誤的描述。而學科

專家及館員所扮演的角色就是在控管資源的品質性。館員也同時需要著錄每筆資源的詮釋資料，最後決定其分類。而系統管理者的角色在於平台的管理，及一些例行性的工作，如資料庫的備份。

而其它功能如 RSS、AJAX NEWS、線上電子字典等功能則是利用 Web 2.0 的 Mashups 的特性，利用一些開放網站服務的 API、Web Feeds (e.g RSS or Atom) 、 Javascript 將該網站的資訊直接引入 SGWS 平台內，以加強入口網站資訊的豐富性。



系統服務架構

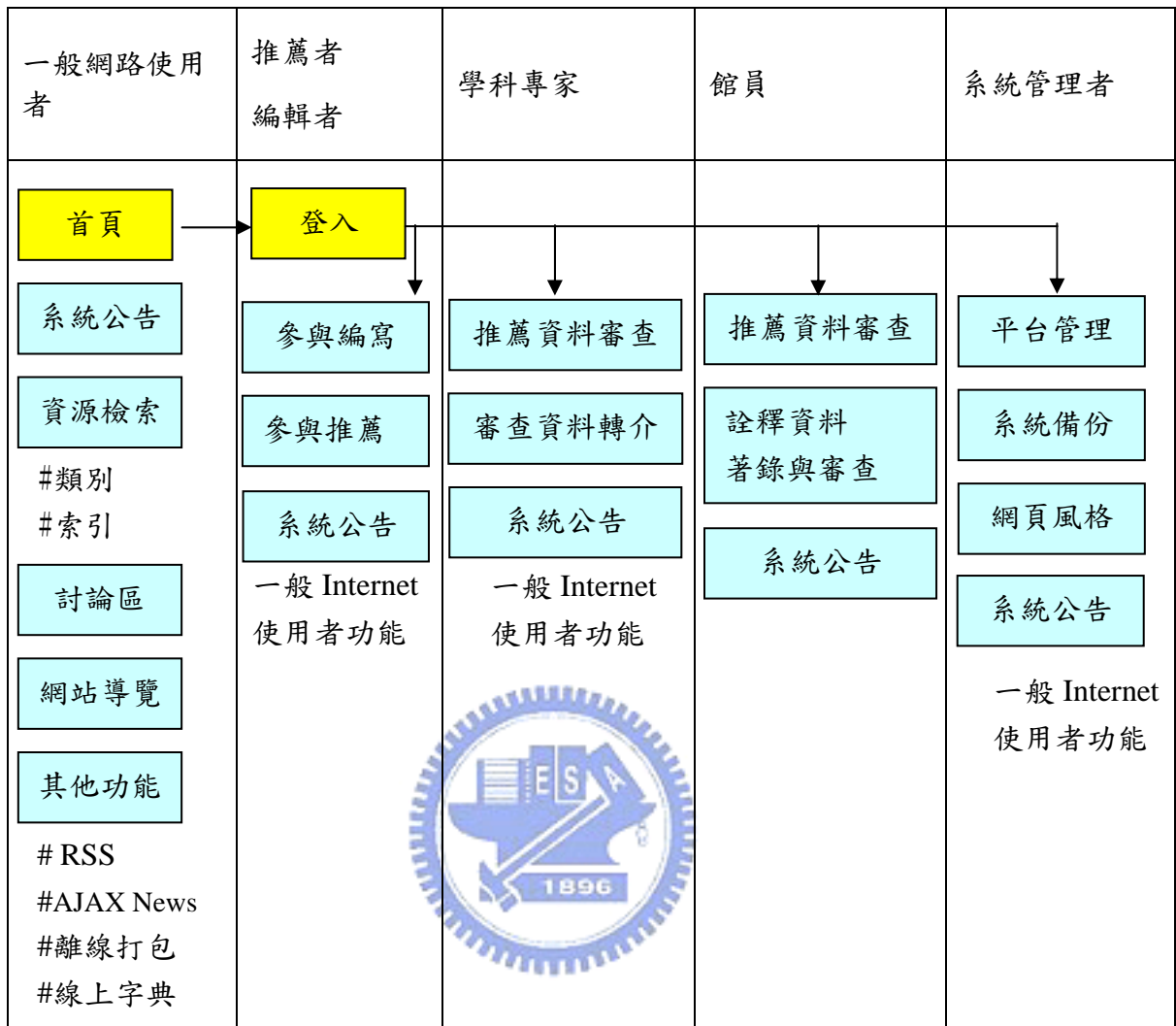


圖 5-2 系統功能圖

5.3 實作過程

研究者將主機環境給架設好後，安裝 ccTiddly Engine (powered by TiddlyWiki ver. 2.0.10 and ccTiddly 0.5.5)，修改程式碼後，實際應用在交通大學圖書館內。截至 2007 年 06 月 22 日，建置好的學院計有 8 個，全部系所總數為 57。使用者經入口頁可選擇功能有「我要搜尋」、「我要分享」、「分類瀏覽」、「相關活動」、「連絡資訊」等。「分類瀏覽」可直接進入相關系所(學院)，直接瀏覽圖書館所製作的資源分類，目前將資源的分類分成國內相關系所、學會組織、研究中心、專業領域、研究中心、研究發表、數位教材、技術報告、學術演講投影片等七個項目，交通大學學科資源網站架構如表 5-1。當然這

些分類架構是十分彈性的，系統仍可依實際需求隨時進行調整。

表 5-1 網站的架構

第一層	第二層	第三層
入口	我要搜尋	簡單檢索
		進階檢索
	我要分享	WebForm 推薦
		Wiki 推薦
	目錄瀏覽	國內相關系所
		學會組織
		研究中心
		專業領域
		研究發表
		數位教材
		技術報告
		學術演講投影片
	相關活動	
	連絡資訊	

(資料來源：<http://140.113.39.236/>。上網日期：2007/06/22)

5.3.1 入口網站

實作之測試站台網址為 <http://140.113.39.236/> (如圖 5-3)，使用者進入網站後，可直接進行檢索資源或至 <http://140.113.39.236/nctu/portal/> 進行分類瀏覽(如圖 5-4)。



圖 5-3 網站入口檢索頁面



圖 5-4 Wiki 入口頁面

5.3.2 認證功能

修改原 ccTiddly 認證模組，更改原認證流程並與圖書館系統認證模組結合。認證功

能的工作流程可參圖 3-2，使用者需經認證後才能參與 SGWS 之互動。例如使用者在 SGWS 時發現錯誤資訊時，則可點選 Wiki 之網址連結(如圖 5-3)，在通過身份認證後到需要更正的頁面上進行修改，認證的程序僅需要輸入右上方認證登錄資訊後即可完成認證程序，系統會自動比對圖書館系統內的有效讀者帳號(如圖 5-5)，正確無誤後，則該使用者即可直接參與修正或進行推薦資源。由於目前所設定的參與社群為交通大學教職員生，若校外人士也發現錯誤資訊時，則可另外透過 E-mail (http://140.113.39.236/nctu/search/recheck.php)的方式回報給管理人員進行處理。



圖 5-5 認證功能

5.3.3 中文化

從電腦的發展史中，可以了解並發現系統的中文化是一個很大的問題。目前主流的電腦系統和軟體大多由國外發展，使用對象也以英語語系者為主，故中文的處理欠佳。一般而言，要完整的解決中文問題，必須從資料的呈現(Display)、傳輸(Transfer)、儲存(Store)、搜尋(Search)四個要素來進行。

本研究於進行時，ccTiddly 0.5.5 版本尚只有英文版可以使用。研究者將介面中文化處理方式是將檔案(tiddlyWiki2.0.10.js)內的特定關鍵詞置換成中文後，再利用 ConverZ 軟體，將之轉成 UTF-8 編碼，即完成中文介面之轉換。而一般以 PHP 語言所撰寫的“許”、“功”、“蓋”的問題，在此由於系統開發的架構，不管是其前端(網頁)

或後端(資料庫)，其 Encoding 的方式都採 UTF-8，故採用 ccTiddly 可正確處理中文資料。

5.3.4 資源的推薦

資源的徵集方式，可透過下列兩種方式進行。第一種方式為利用 Wiki 系統的「推薦資源」功能，直接將要推薦的資訊給加入到系統內，透過 Wiki 系統可採用 Wiki Syntax 進行版面的編排，可提供更為彈性的操作，但使用者仍需額外學習 Wiki Syntax。

第二種方式則是透過現行最為廣泛使用的 Web Form 介面進行(如圖 5-6)，讓使用者在需要的欄位內輸入所需資訊後，按下送出即完成推薦流程，推薦完成的同時，系統亦會主動 E-Mail 通知審核工作人員進行審核，另外為了避免重複推薦、送出錯誤資訊，於推薦時會適時地檢查並提醒是否已存在重複標題與檢查推薦的 URL 是否符合 URL 的格式定義。

SG @ NCTU

Home
圖書館

我要搜尋
我要分享
分類瀏覽
相關活動
連絡資訊

領先的交大、傑出的交大
--- 絕數交大的榮耀

學科資源大分享

請與我們分享優質的學術性網路資源！
分享步驟 1 | 2 | 3 |

1. 先選擇相關系所，以利將資源正確地分享给需要的人
2. 再詳填資源的相關資訊
3. 若發現無法匯上某個網站，請通知我們刪除或更新

要分享資訊給:	--請選擇--
資源名稱(必填):	<input type="text"/>
資源網址(必填):	<input type="text"/> (需符合URL格式) e.g. http://www.lib.nctu.edu.tw/
資源描述(必填):	<input type="text"/>
分享者學號/姓名:	<input type="text"/> <input type="button" value="謝謝您的推薦"/>
相關活動:	只要輸入您的學號或院號，即可參加圖書館館址365之「學科資源大分享」的活動

© NCTU Library 2007

圖 5-6 資源推薦

5.3.5 詮釋資料與搜尋功能(Metadata / Search Function)

搜尋功能的設計區分成簡單搜尋與進階搜尋兩種方式供使用者使用，由於 Wiki 系統導入詮釋資料機制後提供了資料描述的基礎建設，故可改善原搜尋引擎。改善後的搜尋引擎可提供 15 個特定欄位的搜尋方式，圖 5-7 即在說明以關鍵字針對詮釋資料欄位”作品名稱”(Title) 執行檢索所出現的結果，搜尋後的結果，提供了資源的相關說明與連結網址，使用者可直接透過連結網址前往該資源所在地，若要詳細檢視相關內容則可透過「前往 Wiki」直接連結至該 Wiki 頁面，若發現錯誤之描述更可直接與以修改。

The screenshot shows the NCTU Library search interface. At the top, it says "SG @ NCTU" and "領先的交大、傑出的交大 --- 細數交大的榮耀". The main heading is "學科資源大集中" with the slogan "簡單搜尋 1!2!3!". Below this, there are three steps: 1. 輸入關鍵字, 2. 選擇系所, 3. 按下送出. The search form includes a "Search@" dropdown for "Search scope", a "KeyWord:" input field, and "AND" and "ALL" dropdowns. There are "送出" and "Simple" buttons. Below the form, there are radio buttons for search criteria: "作品名稱" (selected), "創作者", "主題", "簡述", "出版者", "貢獻者", "資源類型", "資源格式", "來源", "語文", "關連", and "權限管理". The search results show "您的關鍵詞為Wikipedia" and a table with columns "標題", "描述", and "修正".

標題	描述	修正
Wikipedia	維基百科 (Wikipedia) 是一個多語言版本的自由百科全書協作計劃，已經成為互聯網上最受歡迎的參考資料查詢網站。	前往wiki

@ NCTU Library 2007

圖 5-7 搜尋引擎之改善

5.3.6 審核功能

審核資源的工作流程圖如圖 3-6，審核工作流程可分為兩階段，第一階段由學科專家扮演守門員的角色(圖 5-8)，學科專家僅需決定該資源是否接受。第二階段則由館員接手，除進行審核外，並著錄該筆資源的詮釋資料，最後決定該筆資源的分類。完成分類後，審核過後的資源會自動呈現於瀏覽器上(圖 5-9)。

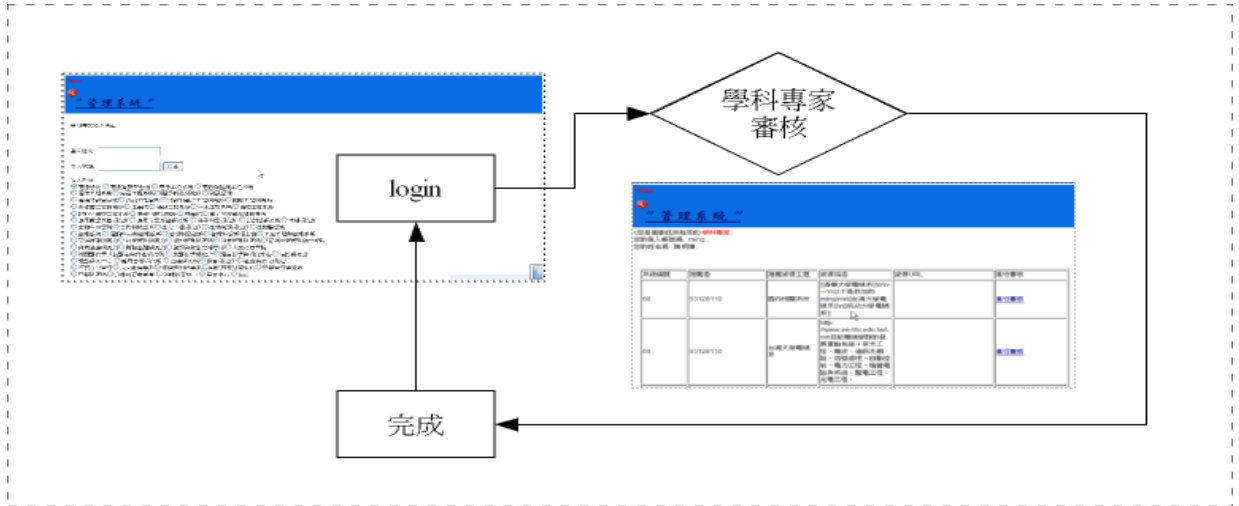


圖 5-8 學科專家審核流程



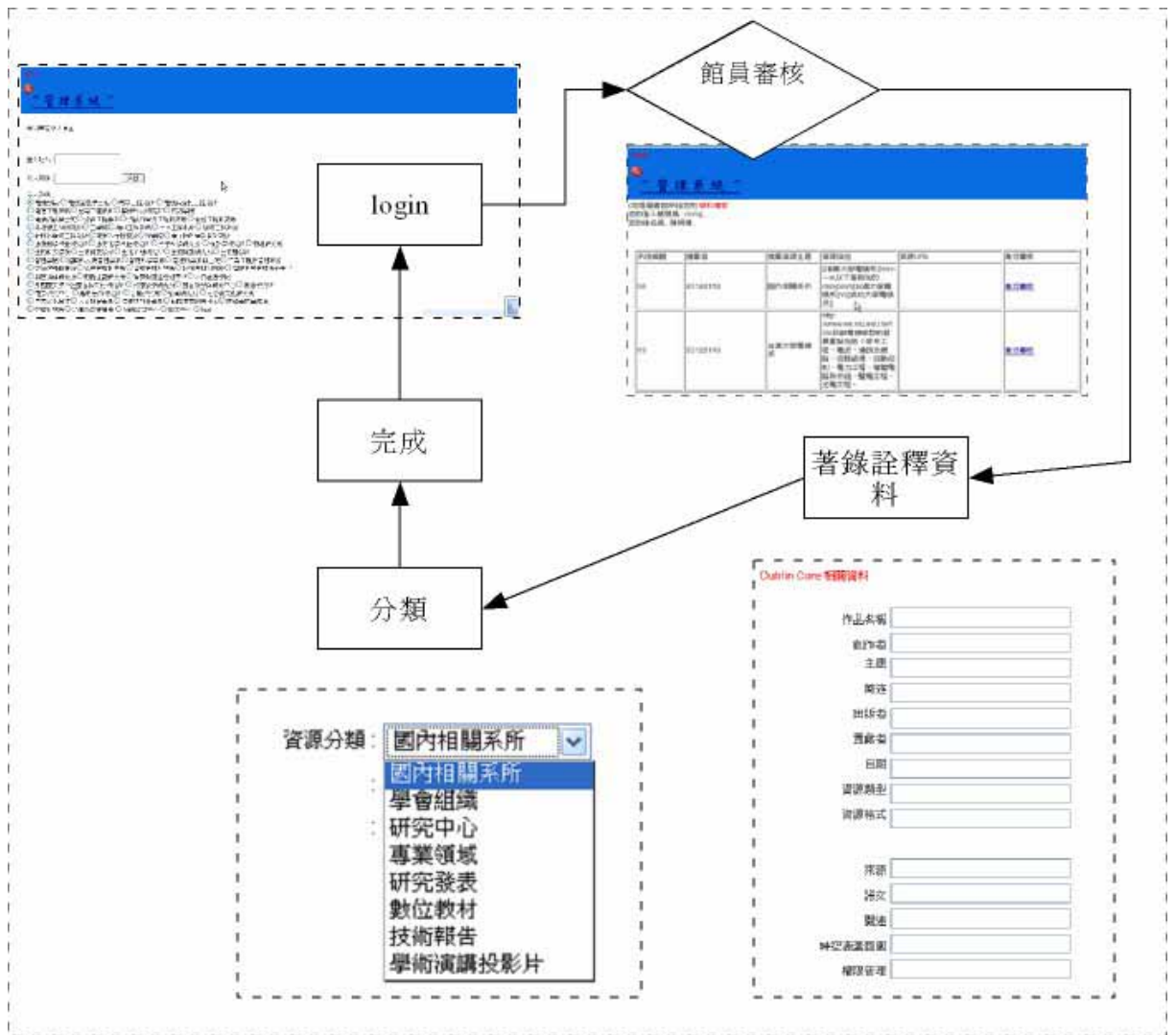


圖 5-9 館員審核流程

5.3.6 混搭(Mashups)

Mashups 的實作是引入 AJAX RSS News、網路電子字典與 Google calendar。AJAX RSS News 這個部份是利用已寫好的 RSS 功能的程式 (<http://140.113.39.236/nctu/portal/includes/rss/AJAX/demo.htm>)，直接嵌入到 Wiki 頁面內即完成 Mashups。該 RSS News 會自動抓取交通大學圖書館最新消息 RSS(<http://blog.lib.nctu.edu.tw/rss.php?blogId=14&profile=rss20>)，自動呈現在 SGWS 頁面內(圖 5-10)。



圖 5-10 AJAX RSS News

網路電子字典功能(圖 5-12)是將開放網站平台的 API 直接引入系統內。僅需要簡單的 Script 就能將該功能引入系統內，豐富網站資源。

```
<html><form method="get" target="_blank"
action="http://url/nctu/includes/dictionary/kimo.php">
<b>Kimo 線上電子字典</b>
<input maxlength=256 size=21 name=Word>
<input type="submit" value="送出">
</form>
</html>
```

圖 5-11 引入 SGWS 系統的 Script

透過簡單的敘述文字(圖 5-11)，就能將現有 Yahoo!奇摩字典功能引入系統內，方便使用者查詢使用。而 Google calendar 的作法也是相似，僅需要前往 Google 註冊該功能，註冊完成後會給註冊者一段 Script，只要將該 Script 嵌入到 Wiki 頁面即完成 Mashups(圖 5-13)，可提供更為豐富的網站資訊。

關閉 關閉其它 編輯 permalink references jump

線上電子字典

W061500, 5 July 2006 (created 5 July 2006)

no tags

Dict.cn

Dict 線上字典

Dict.cn 電子字典,提供最便利的在線上詞查詢和拼寫糾正服務。詞典追求界面簡單,查詢響應即時,解釋意思一目了然。用戶可以輸入英語單詞,英語詞組,漢字,漢語詞語,漢語拼音,英語發音等進行查詢。詞典的人工智能能夠幫助大家迅速找到自己所需要的單詞或詞組。

YAHOO! 字典

Kimo 線上電子字典

Yahoo!奇摩字典提供網友線上的英漢 / 漢英字典查詢,免下載,方便易用,是您翻譯學習的好幫手。

圖 5-12 線上電子字典功能

Metatag
opensearch
pear
php
SG
wiki
web2.0
openid
test

Google 日曆

Ming_open

2007年5月

今日

週日	週一	週二	週三	週四	週五	週六	待辦事項
29	30	1	2	3	4	5	
6	7	8	9	10	11	12	
13	14	15	16	17	18	19	
20	21	22	23	24	25	26	
27	28	29	30	31	1	2	

特區：台北

由 Google 日曆提供 使用

搜尋者:

密碼:

歡迎 ming

最近更新

踴躍順序, 修改日期

- 200705
- 200704
- 200703
- 200702
- [200701]
- 200612

系統採用 TiddlyWiki 2.1.3 and ccTiddly 1.1

[歡迎檢視](#)

圖 5-13 Google calendar 功能

5.3.7 統計功能

SGWS 亦提供統計功能(http://140.113.39.236/nctu/lib/statistics_send)，可統計出網站內通過審核的資源總筆數(圖 5-14)。



起 20070101 ~迄 20070624 [送出]

系所	筆數
電機資訊學士班	2
電子工程系所	15
電機與控制工程系所	15
電信工程系所	13
光電工程系所	10
顯示科技研究所	2
資訊學院	0
電機資訊學士班	0
資訊工程學系	21
資訊科學與工程研究所	30
網路工程研究所	14
多媒體工程研究所	4

圖 5-14 審核統計功能

研究者亦架設 AWstats 軟體，該軟體為功能強大的流量統計工具，分析的數據主要是分析網站伺服器的紀錄檔 (Log files)，故可針對使用者的使用狀況予以分析，可提供日報表、月報表、點擊次數、網頁數、參觀次數...等報表可供參考(圖 5-15)。

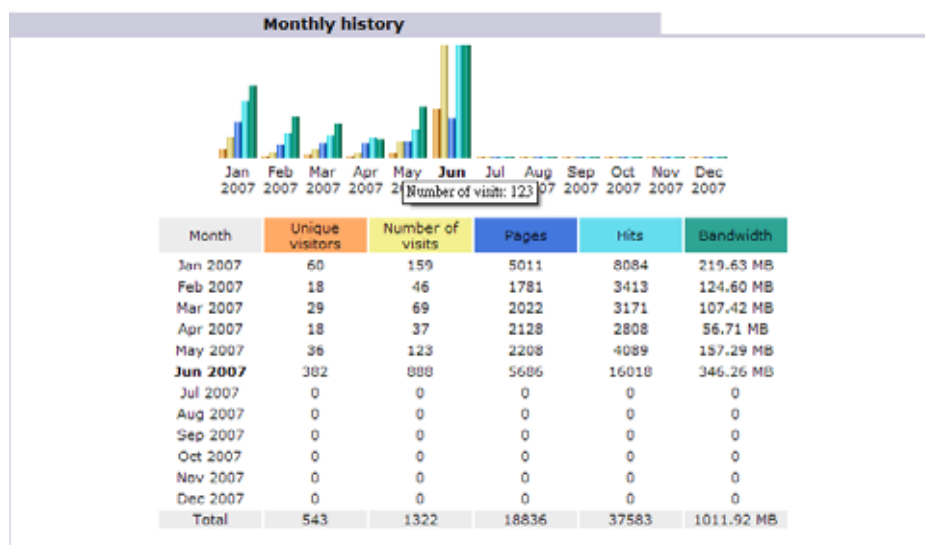


圖 5-15 AWstats 報表

六、研究結論與建議

主題式資源指引是學術圖書館業務的一項特色，由圖書館針對特定的主題範圍有系統地彙整與編纂相關知識。主題式資源指引網站發展至今有一定的成果，但主題式資源指引建置與後續維護成本高，在現階段 Web 2.0 的觀念普及，圖書館應如何應對甚至利用本身所擁有的「最佳社群」，將網站內容建置的主導權部份轉移給「社群」，讓網路資源資料的徵集不再限於少數人(圖書館館員、學科專家)，而由圖書館與讀者雙方共同營造出最符合讀者期待之主題式資源指引網站；除此之外，運用社群的力量亦可促進後續維護工作，透過社群的「監督」協助建置管理人員解決最為困擾的修正網址失效、錯誤內容描述...等問題，不僅減輕工作人員之負擔，更可提供更為正確的資訊。本研究提出下列結論與建議。

6.1 結論

綜合本論文各章研究，本節主要針對現行主題式資源指引網站所面臨困境與以 Wiki 軟體實作主題式資源指引網站所面臨的問題及改善方式歸納整理，結論如下：

一、現行主題式資源指引網站所面臨的問題：

現行主題式資源指引網站所面臨的諸多問題可參考 2.4.3 節討論，主要面臨了資源在哪裡？由哪些人來找尋及整理這些資源？建置好主題式資源指引網站後，如何因應後續維護的問題，例如網址失效 (Broken links)、資源的描述是否過時，收錄的資源是否符合讀者的期望...等。

而採用 Wiki 引擎來管理主題式資源指引網站正可解決部份現行主題式資源指引網站所遭遇的問題，分述如下：

1. 採用本研究所設計的 SGWS 系統，以 Wiki 方式管理產出的內容，可作為替代現行主題式資源指引網站之內容管理系統，並且提供雙向模式共建系統平台，改善原有資料徵集的工作流程。
2. 以「合作共同參與建置」的技術概念，可讓讀者直接參與平台內容主題的維護與推薦，讀者發現失效連結、過時或錯誤的資料描述可即時立即改善，有效解決傳

統主題式資源指引網站較難與讀者進行互動之缺憾。

3. SGWS 提供「雙向」窗口，可讓讀者主動提報及協助資源的發掘與錯誤描述的更正。以交通大學圖書館為個案，SGWS 系統於 2007 年 6 月 6 日釋出供讀者使用，所收錄經審核通過之資源已達 545 筆(2007/01/01~2007/06/22 止)，在系統釋出前，資源的徵集與審核方式是藉由少數工讀生與館員的協助所建立。主題式資源指引網站內容描述的後續維護，因上線時間尚短，目前仍看不到它的效益，但若營造出良好的參與社群，相信採用 Wiki「雙向交流」的模式所運作的主題式資源網站，是可有效的協助資源的發掘、收集與後續的維護作業。
4. 建置 Web-based 的 Wiki 引擎網站，可改善現行主題式資源指引網站建置與維護方式。資源的徵集與後續的維護作業，可提供「雙向交流」的模式，讓使用者可參與網站互動，使用者或管理者僅透過慣用的瀏覽器即可完成網站內容的維護與管理，而不需採用特別的網頁編輯軟體或 CMS 即可完成，也可有效減少教育訓練的相關成本。
5. 系統提供審核統計功能更可得知目前通過審核之總筆數量，透過分析 Apache Logs，更可了解使用者習性，可有效評估系統的使用情況，作為後續分析利用。

二、Wiki 引擎實作主題式資源指引網站所面臨的問題：

採用 Wiki 引擎實作主題式資源指引網站所面臨的問題很多，相關問題點可參考 3.2 節整理。最大的問題點在於圖書館館員首重「品質性」，而 Wiki 本身則強調「開放」與「自由」，任何人都可藉由 Wiki 直接線上更新任何資訊。「品質性」與「開放、自由」有時是相互衝突，要如何在兩者間取得平衡點，是值得深思與討論。本研究設計之 SGWS 有效改善了以 Wiki 引擎實作主題式資源指引網站所面臨的問題：

1. SGWS 利用 Wiki 的特性進行資料的徵集，讀者可自行利用 Wiki 線上直接新增資源，而在其中引入「審核」機制於 Wiki 內，則可有效兼顧品質，雖然這與 Wiki 本身所強調的理念相違背，但站在主題式資源指引網站所收錄的品質性要求則是必要的控管要素之一。
2. SGWS 系統彈性佳，實為一個通用的系統，採用 Wiki 架構的內容管理系統可讓

使用單位依據本身需求、目的、應用直接進行調整。

3. 網路資源的收錄標準多採用 Dublin Core 的方式進行資源的著錄，在 Wiki 中導入詮釋資料對讀者是有益的，可讓資料的搜尋依據不同的角度提供更為深層的搜尋。

系統的設計另一考量即為建置通用且適合使用的系統，故 SGWS 系統仍可適用於不同的應用，如 2.7 節中所提及的社群指引 (Community Guide)、圖書館內部應用 (Wiki Internal Library Wiki)、內容管理系統 (Content Management System)、討論會的 Wiki (Conference Wikis)、知識庫平台 (Knowledge Base)、研討會計劃的 Wiki (Conference planning)……等皆可立即套用於系統內。例如採用 SGWS 建立圖書館的 FAQ 或工作手冊，就是一個很好的實作應用，將圖書館所常面臨的問題，整理成一份線上 FAQ，而當這些資訊需要更動時，則可透過瀏覽器直接進行修改，而不再需要通知資訊人員與以修改，提供快速有效率之方案，FAQ 甚至也可結合 Blog(Wiki + Blog)功能，可讓使用者直接在 FAQ 上回應個人的觀點，提供圖書館後續參考。採用 Wik-based CMS 的相關應用範圍很廣，只要合乎「線上」與「共同參與」特質幾乎都可立即套用，但於應用後是否直能營運的成功，其關鍵因素仍在於相關的政策配套措施，因為系統本身僅為基礎平台，提供一個發揮的場地，使用單位研擬相關政策實刻不容緩(e.g.要如何營造出良好的參與社群，讓參與社群的使用者能樂於參與，要營造那些有特色的主題範圍……等)。

6.2 建議

囿於研究時間與人力等因素，本研究仍有未盡周全之處，系統仍有改善空間，以下提供數點建議供後續研究者參考。

1. 單位內應成立網路資源工作小組，網路資源的蒐集與整理是項新的挑戰，應立即進行研訂相關政策，立即建置「有特色」的主題資源指引網站。工作小組之必要成員應含參考組、編目組、資訊系統組，以及由圖書館所委任的學科專家。館內

成員專責於系統的運作及各式主題資源的建檔描述...等，而學科專家則為其監控品質以確保所推薦之品質。

2. 網路資源工作小組應立即訂立明確的相關政策，主題式資源指引網站面臨的最大困境為建置前後的作業，由文獻中可發覺各館都會嘗試著與其它主題式資源網站合作(例如 DutchESS 專案等)，以求後續永續經營。若以系統的角度來切入，詮釋資料的交換、跨平台檢索、各館集中於一平台共同建置同主題之主題式資源指引網站(例如 CORC 等專案)、引入與其它單位或甚至開放讓全台灣民眾進行參與進行合作共同參與建置...等方面，於技術面是不難辦到的，事實上本研究雛型系統是可配合著既定的政策需求予以改變配合，但相關的配套政策措施才是最大的問題點。
3. Wiki 主題式資源指引網站系統之開發，可朝向提供多種建檔格式及多種顯示及輸出格式方向努力。目前系統僅提供詮釋資料格式建檔，應可考慮加入 USMARC 欄位支援，並提供兩種格式的相互轉換，而使用者也可自行選擇顯示所要呈現的格式。
4. 系統應能支援資料擷取(Information Extraction)且完成自動編目，並將待編目之網頁自動呈現於下半部，方便工作人員檢視，以節省檢閱時間與人力。本研究之系統目前僅能自動擷取網頁內嵌的詮釋資料(Metatag)。
5. 系統應能提供讓工作人員自行定義較常出現的字彙(例如預設值之功能)，以減輕重複性的輸入資料。
6. 系統應可提供相關主題式資源指引網站 (Subject Guide) 的自動建立，提供圖書館更深度的服務。也就是說管理者，可將相似的主題或不同主題，利用系統自動地建立這些不同分類的主題指引，不需再經另一次的加工，而能產生不同主題資源指引網站供使用者參考。
7. 標準之遵循，有助於減少相關問題產生：

- (1) IETF RFC2731 標準：網頁遵循 IETF RFC2731 方式描述網頁詮釋資料，則系統在擷取詮釋資料時就能更為正確，如此便能減少編目人員的輸入作業。

- (2) Wiki Syntax 標準：Wiki 引擎若採用統一的 Wiki Syntax 標準，則往後若面臨系統的轉換時，則可直接進行轉換，不需考慮 Wiki Syntax 的不同而必須額外處理。
- (3) CSS 標準：CSS 是 HTML 4.0 規格提出時，一併提出的新規格，而目前所使用的瀏覽器，其通用的兩大派系為 Microsoft 所用的 Internet Explorer 核心，另一種為 Mozilla 基金會開發倡導的 Gecko 核心，此二套瀏覽器程式核心在解讀 CSS 時會有些許的差異。若各瀏覽器支援 CSS 的標準統一，則能減少資訊人員對程式碼的重新編寫。
- (4) OPENID 的支援(<http://openid.net/>)：事實上，不管圖書館採用那種內容管理系統來管理網站內容，而其認證模組又要與現行圖書館系統結合，就認證模組而言仍需自行客製化，這增加了相關工作人員的負擔，認證模組可考慮採用現行 OPENID 的方式，如此便可減少相關成本的投入。



參考文獻

1. Gorman, M., "The five laws of library science then & now." School Library Journal, 1998. **44**(7): p. 20~24.
2. Netcraft. *November 2006 Web Server Survey*. 2006 [cited 2006 11/06]; Available from: http://news.netcraft.com/archives/web_server_survey.html.
3. 卜小蝶、鍾季倫、郭佩宜, 「主題式資源指引網站之發展初探」. 國家圖書館館刊, 2005. **94**(2): p. 1~25.
4. *OpenID*. 2007 [cited 2007 05/30]; Available from: <http://openid.net/>.
5. 胡述兆總編輯, 圖書館學與資訊科學大辭典. 1995, 漢美: 台北市. p. 2079.
6. Courtois, M.P., "Was this guide helpful? Users's perceptions of subject guides". Reference Services Review, 2005. **33**(2): p. 188~196.
7. Wales, T., "Library subject guides:a content management case study at the Open University,UK". Program, 2005. **39**(2).
8. Lorcan Dempsey, T.G., and Michael Day, "International Information Gateway Collaboration Report of the First IMesh Framework Workshop". D-Lib Magazine, 1999. **5**(12).
9. Consortium, D. *What is a subject gateway?* 1998-2000 [cited 2006 10/18]; Available from: <http://www.desire.org/html/subjectgateways/subjectgateways.html>.
10. Koch, T., "Quality-controlled subject gateways: definitions, typologies, empirical overview". Online Information Review, 2000. **24**(1): p. 24~34.
11. 柯皓仁, 「電子圖書館實施經驗談--以交通大學圖書館為例」. 書苑, 2001. **47**: p. 10~33.
12. McDonald, S.D.a.P., "The Internet and Collection Development: Mainstreaming Selection of Internet Resources". LRTS, 1995. **39**(3): p. 275-290.
13. 詹麗萍, 「電子資源與圖書館館藏發展」. 2005: 麥田.
14. consortium, D. *DESIRE Information Gateways Handbook*. 2000 [cited 2006 10/25]; Available from: <http://www.carnet.hr/CUC/cuc2000/handbook/contents.html>.
15. Chepesiuk, R., "Organizing the Internet: The "Core" of the Challenge." ERIC, 1999. **30**(1): p. 60~63.
16. OCLC. *Cataloging Internet Resources: A Manual and Practical Guide*. 1998 [cited 2006 11/24]; Available from: <http://www.oclc.org/support/documentation/worldcat/cataloging/internetguide/>.
17. 曾秋香, 「電子資源合作編目計劃之研究 :以 CORC 為例」. 國家圖書館館刊, 2002. **91**(1): p. 59~78.
18. OCLC. "OCLC Invites Libraries to Help Build the Cooperative Online Resource Catalog." Computers in Libraries, 2000. **20**(3).

19. MaoJun, "Bufliding Subject-Based Information Gateway (SBIG) with ROADS". Library and Information Service 2001(11).
20. 羅思嘉, 「數位圖書館資源組織與整理: 機讀編目格式與 Dublin Core」. 大學圖書館, 1999. 3(2): p. 50-68.
21. Calhoun, K. *CORC at Cornell Project: Final Report*. 1999 [cited 2006 11/09]; Available from: <http://www.library.cornell.edu/corc/corc-final.htm>.
22. Rebecca Jackson, L.J.P., "Internet Subject Guides in Academic Libraries". Reference and User Services Quarterly, 2004. 43(4): p. 327-335.
23. Howarth, L.C., "Metadata Schemas for Subject Gateways". International Cataloguing and Bibliographic Control, 2004. 33(1): p. 8-12.
24. Northrup, L.A., "Creating Easy to Update Subject Guides Without Using a Database." College & Undergraduate Libraries, 2006. 13(1): p. 53~57.
25. 杜毓真、陳明燁、曾婉珍、楊承嘉、簡燕華, 「數位參考服務平台 ResearchGuide 實用性之初探」. 2005, 國立交通大學電機資訊學院在職專班數位圖書資訊組: 新竹市. p. 1~12.
26. 中華民國圖書館學會, 大學圖書館設立及營運基準. 2004年7月28日.
27. Achterman, D., "Making Connections with Blogs and Wikis." CSLA Journal, Fall2006. 30(1): p. 29~31.
28. Lackie, R.J., "WEB 2.0 and Its Technologies for Collaborative Library Communication". MultiMedia & Internet@Schools, 2006. 13(6): p. 9~12.
29. O'Reilly, T. *What Is Web 2.0*. 2005 [cited 2006 11/20]; Available from: <http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>.
30. Wikipedia, c. *Web 2.0*. 21 February 2007 13:21 UTC [cited 23 February 2007 01:36 UTC]; Available from: http://en.wikipedia.org/w/index.php?title=Web_2.0&oldid=109805701
31. 梁鴻翔. *Library 2.0 是什麼*. 2006 [cited 2006 11/24]; what is library2.0 and web2.0].
32. eConsultant. *There are 1200+ Web 2.0 sites/services/links in 50+ categories*. 2006 [cited; Available from: <http://www.econsultant.com/web2/>].
33. Keven. *Web2.0 應用型態*. 2006 [cited 2006 11/24]; web2.0 application]. Available from: <http://my.donews.com/keven/2006/05/13/statisticsbygoogletrends/>.
34. Wikipedia, c. *Wiki*. 21 February 2007 19:41 UTC [cited 23 February 2007 01:47 UTC]; Available from: <http://zh.wikipedia.org/w/index.php?title=Wiki&oldid=3555522>
35. Ebersbach, A., Wiki :/Web collaboration, ed. M.G.a.R.H. Anja Ebersbach. 2005: Berlin.
36. 中央研究院資訊科學研究所鑄, 「Wikipedia: 一場有著開放精神的社會革命,正在進行中」, in Linuxpilot 國際中文版. 2006. p. 36~39.
37. Wikipedia, c. *Wiki engine*. 16 February 2005 17:27 UTC [cited 23 February 2007

- 01:49 UTC]; Available from:
http://en.wikipedia.org/w/index.php?title=Wiki_engine&oldid=17207173
38. Wikipedia, c. *WikiWikiWeb*. 22 February 2007 00:10 UTC [cited 23 February 2007 01:38 UTC]; Available from:
<http://en.wikipedia.org/w/index.php?title=WikiWikiWeb&oldid=109943835>
39. Wikipedia, c. *TiddlyWiki*. 18 February 2007 05:01 UTC [cited 23 February 2007 01:35 UTC]; Available from:
<http://en.wikipedia.org/w/index.php?title=TiddlyWiki&oldid=109002018>
40. Ruston, J. *CCtiddlywiki History*. 2006 [cited 2006 11/22]; Available from:
<http://cctiddly.sourceforge.net/#History>.
41. Farkas, M. *How Libraries Can Use*. 2006 [cited 2007 01/11]; Available from:
http://www.libsuccess.org/index.php?title=How_Libraries_Can_Use.
42. Frumkin, J., "The Wiki and the digital library". *OCLC Systems & Services*, 2005. **21**(1): p. 18-22.
43. Fichter, D., "Using Wikis to support online collaboration in libraries". *information outlook*, 2006. **10**(1).
44. Matthies, B., "Using A wiki to Enhance Library Instruction". *Indiana Libraries*, 2006. **25**(3): p. 32-34.
45. Kolbas, J., *Wikis: The Tools for Information Work and Collaboration*. 2006: Chandos Publishing (Oxford) Ltd.
46. Stephens, M., "Wikis." *Library Technology Reports*, 2006. **42**(4): p. 52~57.
47. Lewis, B.C.a.P.H., "WikiWikiWebs:New Way to Communicate in a Web Environment". *Young Adult Library Services*, 2006. **4**(3): p. 33~43.
48. E. Bressler & E. Grantham, 蔡., 丁惠民譯, *社群行銷/Communities of commercebuilding Internet business communities to accelerate growth*. 2002: 麥格羅希爾.
49. *Community Model*. 2007 [cited; Available from:
<http://ecommerce.hostip.info/pages/231/Community-Model.html>.
50. Desilets, A. "Are Wikis Usable". in *NRC.CNRC*. 2005. San Diego, California, USA: International Symposium on Wikis.
51. Meredith. *Wiki: The Ultimate Tool For Online Collaboration*. 2006 [cited 2006 11/22]; Available from:
<http://meredith.wolfwater.com/wordpress/index.php/2006/07/07/wiki-webcast-youre-invited/>.
52. Farkas, M. *Wiki: The Ultimate Tool For Online Collaboration*. 2006 [cited 2006 11/2]; Wiki: The Ultimate Tool For Online Collaboration]. Available from:
http://www.sirsidynixinstitute.com/seminar_page.php?sid=66.

附錄一

```
< member_login.php.php files> page 1 //登入頁面
<META http-equiv="Content-Type" content="text/html; charset=UTF-8">
<?php
    //includes
    include_once("includes/header.php");
    include_once("includes/print.php");

    //Logout
    if( isset($_GET['logout']) && $_GET['logout']==1 )
    {
        setcookie('txtUserName',$_POST['cctuser'],$tiddlyCfg['pref']['cookies'],"/");
        setcookie('pasSecretCode',$_POST['cctpass'],$tiddlyCfg['pref']['cookies'],"/");
        header("Location: index.php"); //redirect to index.php to refresh
    }

    //reqLogin
    //set cookie for login
    include_once("includes/lib_functions_userlogin.php");
    $valid = check_password($_POST['cctuser'],$_POST['cctpass']);
    if ($valid==1) {
        setcookie('txtUserName',$_POST['cctuser'],$tiddlyCfg['pref']['cookies'],"/");
        setcookie('pasSecretCode',md5($_POST['cctpass']),$tiddlyCfg['pref']['cookies'],"/");
        header("Location: index.php"); //redirect to index.php to refresh
    }

    //print login box if not logged in or not in the right
    group//print login box if not logged in or not in the right
    $user = user_create();
    if( !$user['verified'] ) //if not logged on, display login screen
    {
?>
<html><head>
<link href="http://www.lib.nthu.edu.tw/fontstyle.css" rel="stylesheet" type="text/css">
</head>
<body>
<p>&nbsp;</p>
<p> 歡迎來學科資源共建共享系統</p>
<p><br>
    本知識庫各資源之描述都可供圖書館所指定的學科專家及本館有效讀者自由編輯。</p>
<p> 您在本站學習各類學科知識之餘，亦可以參與本網站的建置、改善本網站的文獻的質素。 </p>
<p> 請輸入,您在圖書館系統內的使用帳號及密碼<br>
</p>
<form name="form1" method="post" action="member_login.php">

<?php print $ccT_msg['loginpanel']['username']?>
<input type="text" name="cctuser"><br>
<?php print $ccT_msg['loginpanel']['password']?>
<input type="password" name="cctpass">
<input type="submit" value="<?php print $ccT_msg['loginpanel']['login'] ?>" name="Submit">
</form>
</body></html>
<?php
    exit("");
}
```


附錄二

< lib_functions_userlogin.php files> page 1

<? //認證用程式碼

```
$RNO=$_COOKIE['txtUserName']; //get cookie
```

```
$PWD=$_COOKIE['pasSecretCode']; //get cookie
```

```
function check_password($userid, $password)
```

```
{
```

```
    $ServerIP = "ip address";
```

```
    $ServerPort = "port no.";
```

```
    $fp = fsockopen($ServerIP, $ServerPort);
```

```
    if(!$fp) //Something wrong with the password server
```

```
    { //mail webadm to solve it!
```

```
        error_log("Password server down!", 1, "Webadm");
```

```
        return 4;
```

```
    }
```

```
    $query = "4xxxSELF";
```

```
    $query .= $userid;
```

```
    $query .= str_repeat(" ", 12 - strlen($userid));
```

```
    $query .= $password;
```

```
    $query .= str_repeat(" ", 16 - strlen($password));
```

```
    $query .= chr(13).chr(10);
```

```
        fputs($fp, $query, 38);
```

```
    $result = fgets($fp, 64);
```

```
    switch(substr($result, 2, 3))
```

```
    {
```

```
        case 2:
```

```
            return 1;
```

```
            break;
```

```
        case 8:
```

```
            return 2;
```

```
            break;
```

```
        case 18:
```

```
        case 21:
```

```
        case 23:
```

```
        case 24:
```

```
        case 25:
```

```
        case 26:
```

```
        case 27:
```

```
        case 28:
```

```
            return 3;
```

```
            break;
```

```
        default:
```

```
            return 4;
```

```
            break;
```

```
    }
```

```
}
```

```
//-----check ok do it
```

```
if ($RNO) :
```

```
$r_code=check_password($RNO, $PWD);
```

```
if ($r_code==1) {
```

```
    session_register("LOGIN");
```

```
    $LOGIN = 1;
```

```
// for ccTiddly check
```

```
$n = $_COOKIE['txtUserName'];
```



< lib_functions_userlogin.php files> page 2

```
$p = $_COOKIE['pasSecretCode'];
$user = "$n";
$passwd = "$p";
//get a and b array
$a= array($user);
$b= array($passwd);
//$mix = array_combine($a1,$a2); //combine array but it for php5
//get new array for user_check
for ($i=0;$i<count($a);$i++) {
    $user_check[$a[$i]] = $b[$i];
}
$tiddlyCfg['user'] =$user_check;

//-----check not ok do this
} else {
//erase cookie
setcookie("txtUserName");
setcookie("pasSecretCode");
$tiddlyCfg['user'] = array("xxxxxx"=>"xxxxx"); // if check not ok ,just assign the account info.
}
else:
endif;
?>
```

