

國立交通大學

資訊學院 數位圖書資訊學程

碩士論文

文件注釋系統設計與開發之研究

The Design and Development of Web-based Annotation System



研究生：林易賢

指導教授：黃明居 博士

中華民國一〇〇年一月

文件注釋系統設計與開發之研究
The Design and Development of Web-based Annotation System

研究生：林易賢

Student：Yi-Hsien Lin

指導教授：黃明居

Advisor：Dr. Ming-Jiu Hwang



Master of Science

in

Digital Library

January 2011

Hsinchu, Taiwan, Republic of China

中華民國一〇〇年一月

文件注釋系統設計與開發之研究

研究生：林易賢

指導教授：黃明居

國立交通大學

資訊學院

數位圖書資訊學程碩士班

摘要

閱讀文章時，閱到難懂的內容時，經常需要尋找更多的資料來輔佐以理解文章內容，如果能透過工具將已閱讀過的讀者所留下的見解與記錄，應能讓後來的閱讀者能更快的理解內容，內容標誌(Content Markup)的主要工作就是在描述個別的內容，如對內容的理解與感受。本研究透過內容標誌與文件注釋的概念，建立一個系統可以讓使用者進行文件處理、知識結構的建立、內容注釋與對映，讓文件之間建立起關聯性，讓使用者透過內容標誌可以直接閱讀相關段落，並找尋到更多相關的資訊。此外，使用者也可以在內容上進行註記，並分享給予其他使用者，系統並依使用者的操作進行社群的分類。使用者在進行文件對映後，使用者就能直接透過知識結構來進行相關文件的查詢，更快的找到相關的文件。

關鍵字：網頁注釋、內容對映、知識結構、內容標誌

The Design and Development of Web-based Annotation System

Student: Yi-Hsien Lin

Advisor: Dr. Ming-Jiu Hwang

Degree Program of Computer Science

National Chiao Tung University

Abstract

We always search more information to assist us to understand articles when we are reading and encounter difficult subjects. If there are tools and systems can keep information and comment from previous reader. It may be helpful to later reader to understand articles easier and quicker. Main purpose of Content Markup is describing individual content, including comprehension and feeling of content. This research establishes one system through concept of Content Markup and Document Annotation to provide users document processing, knowledge structure establishment, content markup and mapping to build connections between documents. Users can read related section by Content Markup and find more related information easier. Furthermore, users can write comments for articles content and share to other users. System also classify users into communities.

Keyword: Web Annotation, content mapping, Knowledge structure, Content Markup

誌 謝

兩年多來工作與念書兩邊忙碌，特別感謝黃明居老師在各方面的教導與指教，也總是在快無力時加以鼓勵，讓我更有動力持續進行研究；此外也要感覺相互砥礪的同學，尤其是揚喬與姿伶同學，總是時時相互詢問論文進度；還有正霖與峪嵐學長在論文研究上心路歷程的分享，在論文完成的過程中，接受過許多人的幫助，在此均獻上由衷的感謝之意。

研究所學習之途即將告一段落，但人生學習的路程依舊，往後依然會持在校所學，持續努力，讓自己不斷成長。



目 錄

中文摘要.....	i
英文摘要.....	ii
誌 謝.....	iii
圖 目 錄.....	vi
表 目 錄.....	viii
第一章、緒論.....	1
1.1.研究動機與目的.....	1
1.2.研究內容.....	2
1.3.研究範圍與限制.....	2
1.4.研究流程.....	3
1.5.論文大綱.....	4
第二章、文獻回顧.....	5
2.1.文獻回顧.....	5
2.1.1.超文件.....	5
2.1.2.文件之間的關聯性.....	6
2.1.2.內容標誌.....	8
2.1.3.中文文獻系統.....	10
2.1.4.文件內容注釋.....	12
2.2.使用的程式工具.....	16
2.2.1.Javascript.....	16
2.2.2.ExtJS 3.0.....	18
2.2.3.Python 2.5.....	19
2.2.4.Graphviz 圖形表示工具.....	20
第三章、系統分析與設計.....	21
3.1.系統設計概念.....	21
3.2.設計架構與模式.....	21
3.3.系統模組設計.....	23
3.3.1.文件工具模組.....	24
3.3.2.知識架構處理模組.....	26

3.3.3.內容對映模組.....	26
3.3.4.注釋模組.....	29
3.3.5.個人收藏模組.....	30
第四章、系統功能展示.....	31
4.1.系統實作環境.....	31
4.2.系統實作結果.....	32
4.2.1.文件功能.....	32
4.2.2.知識結構功能.....	34
4.2.3.文件對映功能.....	36
4.2.4.文件注釋功能.....	43
4.2.5.個人收藏管理功能.....	45
4.2.6.文件關聯圖.....	46
第五章、結論與未來研究.....	48
5.1.研究結論與貢獻.....	48
5.2.未來研究.....	49
參考文獻.....	51
附錄一 Graphviz 圖形類型表.....	54



圖目錄

圖 1-1 研究流程圖	3
圖 2-1 文獻整理	5
圖 2-2 不同版本文件內容示意圖	7
圖 2-3 不同版本文件內容比較圖	7
圖 2-4 內容連結示意圖	8
圖 2-5 中文文獻系統	10
圖 2-6 知識結構與文件關聯圖	11
圖 2-7 傳統注釋內容	13
圖 2-8 資訊找尋時間比較圖	13
圖 2-9 注釋系統分類圖	15
圖 2-10 注釋方式表示圖	15
圖 2-11 ExtJS 架構圖	19
圖 3-1 系統概念圖	21
圖 3-2 三層式架構圖	22
圖 3-3 系統MVC設計架構圖	23
圖 3-4 系統設計架構圖	24
圖 3-5 文件工具模組流程圖	25
圖 3-6 知識架構處理流程圖	26
圖 3-7 內容對映模組流程圖	27
圖 3-8 網頁文字定位流程圖	28
圖 3-9 注釋模組流程圖	29
圖 3-10 個人收藏模組流程圖	30
圖 4-1 系統實作流程圖	31
圖 4-2 系統功能架構圖	32
圖 4-3 文件上傳畫面	33
圖 4-4 文件上傳結果	33
圖 4-5 文件載入功能	33
圖 4-6 文件閱讀區塊	34
圖 4-7 關聯名單顯示	34

圖 4-8 知識樹新增畫面	35
圖 4-9 知識樹節點新增畫面	35
圖 4-10 知識樹內容	36
圖 4-11 新增文件對映	37
圖 4-12 節點對映文件	37
圖 4-13 文件對映節點	38
圖 4-14 心經法月版本對映	39
圖 4-15 心經玄奘版本內容	39
圖 4-16 通序節點對映心經法月版	40
圖 4-17 通序節點對映心經玄奘版	40
圖 4-18 使用者 A 的對映內容	41
圖 4-19 使用者 B 的對映內容	41
圖 4-20 典雅類唐詩對映	41
圖 4-21 對映詩品本文	42
圖 4-22 節點載入文件	43
圖 4-23 節點對映文件段落(1)	43
圖 4-24 節點對映文件段落(2)	43
圖 4-25 加入節點注釋	44
圖 4-26 文件內容注釋	44
圖 4-27 顯示節點注釋	45
圖 4-28 顯示文件內容注釋	45
圖 4-29 管理我的收藏	46
圖 4-30 載入個人收藏	46
圖 4-31 文件關聯圖	47

表 目 錄

表 2-1 文章數位化後的訊息表.....	8
表 2-2 網頁注釋工具比較表.....	16
表 2-3 Javascript 與 Java Applets 比較表.....	17
表 2-4 ExtJS 函式庫.....	18



第一章、緒論

本章描述本研究的動機與目的及希望解決之問題。1.1 節說明研究動機與本研究期望達成的目標；1.2 節為研究的內容說明；1.3 節為研究的範圍與限制；1.4 節為研究的流程；1.5 節說明本論文各章節的內容架構。

1.1. 研究動機與目的

文件資料閱讀的過程中，如遇到艱深難懂的內容時，如古文或外文文獻內容，經常再尋找許多資料來輔佐閱讀，找到的資料也都需要重新閱讀，進而理解，如果在過程中可以吸收其他人閱讀過所留下來的想法與解釋，應能讓自己更快速的理解文件內容；此外，每一個人對同一件事物的看法不一定相同，如何將每一個人對文件的見解與不同詮釋記錄下來，是一個值得研究的課題；謝清俊教授所提到的內容標記(Content Markup)的主要工作就是在描述個別內容，以文章來說就是對內容的理解、感受，詮釋的部分，而這個標記是否可以再讓不同的文件產生關聯，讓內容標記也可以讓文件之間搭起一座聯繫的橋樑，讓閱讀者透過內容標記連結閱讀相關段落，因為文章存在電腦以後，就不是一個孤島，它的內容、字串必須與其他文章有點關係[1]。

現在許多文件資訊均數位化並透過網際網路流通，網頁也變成資訊顯示的重要媒介，網路也成為資訊分享的平台，Web 2.0 的概念應運而生。因此，如何在這分享的環境中建置一個可以讓使用者容易使用，並對文件進行內容標記的系統，讓資料不再只是資料本身，並透過此系統讓資料與資料之間的連結更為密切，讓使用者能更快的了解內容與尋找更多相關的資料，為本研究主要動機所在。

因此，本研究的目的包括：

1. 設計與建置一個文件注釋平台，並探討相關技術與功能。
2. 讓人文學者或非資訊背景相關使用者方便為文章進行內容注釋與內容標記的動作。
3. 如何利用內容標記連結不同的文章，讓使用者可以透過文章之間的關係，尋找到更多的相關資料。

1.2. 研究內容

針對本研究目的，本研究內容包括：

1. 如何利用網頁技術，讓使用者可以在網頁文字上進行注釋，系統如何準確的擷取使用者所選取的範圍並加以記錄。
2. 如何將內容標誌的概念與注釋技術加以結合，擷取到不同使用者對同一篇文章的不同詮釋與見解。
3. 是否能透過內容標誌更快的找尋到相關的文章段落。
4. 如何透過已建立好的連結，顯示出各物件之間的關係。

1.3. 研究範圍與限制

受限於研究者可以取得資源有限，為避免研究問題過於複雜，且保持研究問題的單純性，本研究的限制如下：

1. 本研究將以網頁(HTML)文字注釋做為研究目標，尚未擴及至多種線上文件種類。
2. 系統開發環境以網際網路作業環境為主，因係以 Javascript 為主要開發工具，目前市面所擁有的網頁瀏覽器(Browser)並未都支援正規的 Javascript 程式內容，所以系統閱覽需限定在 Microsoft Internet Explorer。Mozilla Firefox 或 Google Chrome 需安裝 IETab 元件才能使用。
3. 系統內容以使用者難以閱讀的古籍內容或外文文獻為內容標的；本研究只針對系統面進行深入研究，並不包含文件內容本身。
4. 網頁內容在上傳後就不能變更，因為內容變更後會造成系統定位錯誤。

1.4. 研究流程

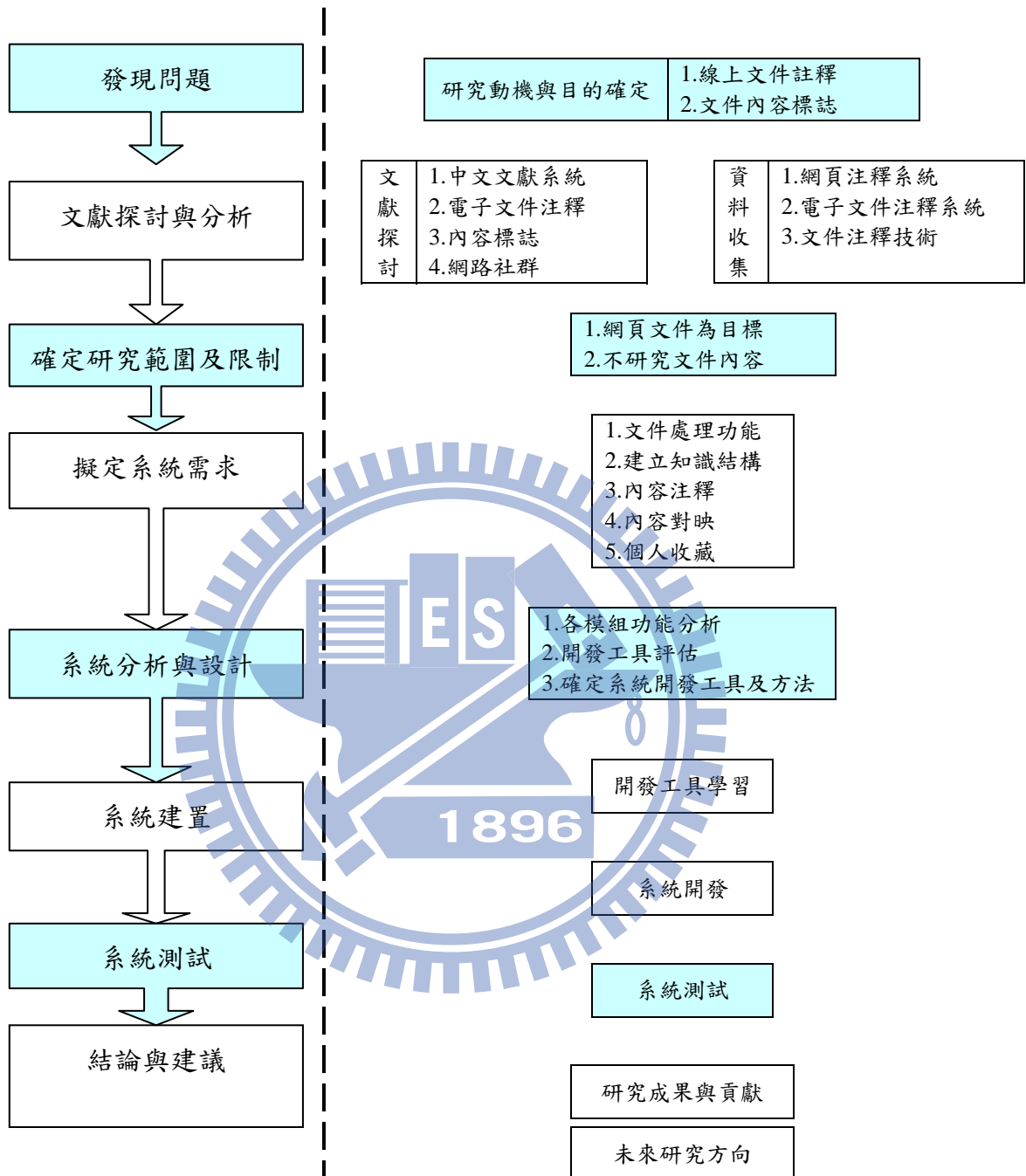


圖 1-1.研究流程圖

1.5. 論文大綱

本論文主要分為五個章節，整理架構流程如下詳述：

1. 第一章為緒論，透過研究動機與目的，說明所遇到的問題並提供可行的解決方案，並訂定研究限制及研究流程。
2. 第二章為文獻回顧與系統功能需求描述，介紹超文件、中文文獻系統、電子文件注釋工具的特性，並描述系統功能需求與所使用的工具程式。
3. 第三章為系統分析與設計，描述系統架構、資料處理方法，並解釋各模組所處理的資料。
4. 第四章為系統展示，說明系統各項功能。
5. 第五章為結論與未來研究方向，為本研究之總結，說明本研究之貢獻，並提出未來可繼續研究之議題。



第二章、文獻回顧

本章內容分為三個部分，2.1 節就研究所需，說明相關文獻；2.2 節為建置本系統所應用的系統工具。

2.1. 文獻回顧

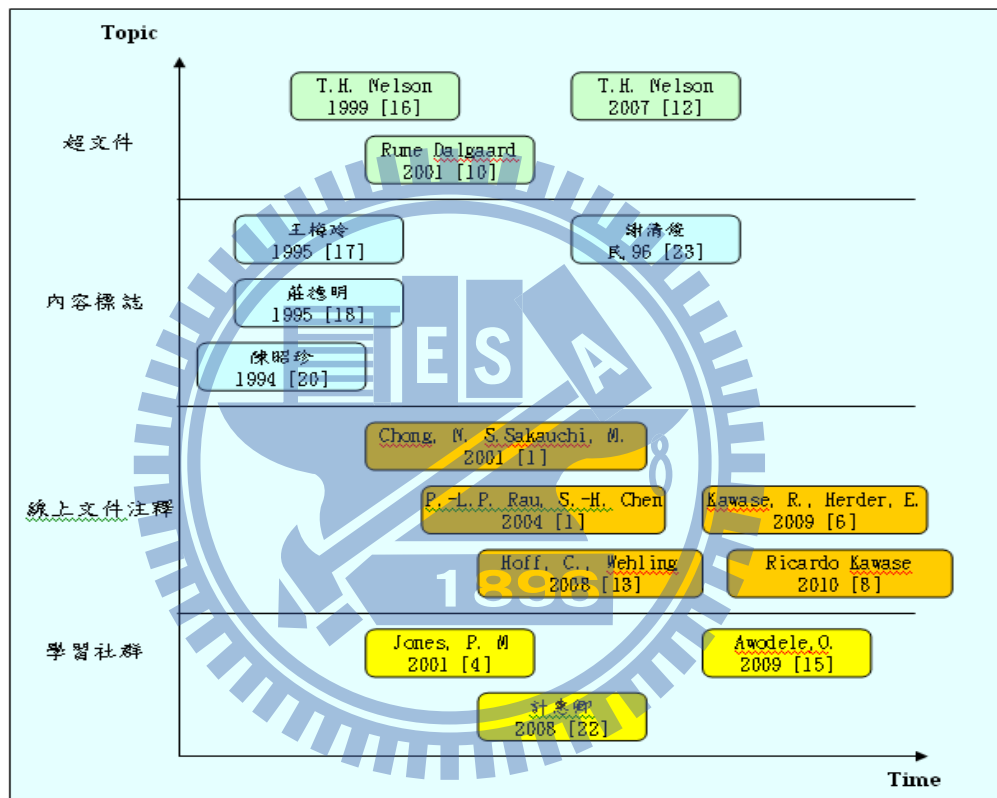


圖 2-1.文獻整理

2.1.1. 超文件

文件(Document)是經由國際標準組織(International Standard Organization, ISO)定義為：「是傳達作者思想之物件，無論呈現的媒體為何，其主要目的乃在正確的傳輸、溝通及儲存作者之概念」[3]。所以作者透過文字建立文件來表達當時所要傳達的內容，但語言文字只是表達抽象概念的媒介[3]；超文件(Hypertext)是1960年代所發展出來運用於網際網路傳播資訊的概念[4]，構想是起源於1945年Vannevar Bush所設想之MEMEX系統，不過首先使用此名詞的是T.H.

Nelson。超文件系統利用電腦將文件資訊解構成「文件單元」(Text Chunk)，並提供文件單元連結，讓使用者可以任意連結相關資訊[5]。

超文件的概念取自於人類認知模式，人類在學習的過程中藉由聯想來與其他事物產生連結，也就是觸類旁通之行為[6]。「連結」(Link)可視為人類的本能，也是資訊的本質，單一或混亂的資訊是無法建立知識體，只有透過資訊的連結才能發揮效用[7]；此概念也被視為文本性(Textuality)的新類型，打破了文字是以線性輸出為主要的概念，被認為可以帶來動態(Dynamic)、解構(Deconstructed)與分散(dispersed)的文字內容。

連結的非線性結構也是超文件的一大特性，它打破了文字是以線性輸出為主的慣例[8]，也有其優缺點。超文件主要就是將原文件「解構」後在電腦上「重組」，而重組的好壞將會影響讀者的認知與學習，而系統設計者則是利用節點(Node)與連結(Link)來構成超文件系統的基本概念[9]，而連結為超文件系統的核心，如果連結建置不當，等同於重組文件不當而影響使用者學習，讓超文件沒有達到非線性的特性而產生更多的限制。

2.1.2. 文件之間的關聯性

一般在論文研究的過程中，會尋找許多相關領域的文章，通常會利用參考文獻的資訊來建立文件之間的關聯性，但這關聯性的單位為整篇文章，如果沒有透過參考文獻，文件之間的關聯性就十分薄弱；而網頁之間則是透過超連結(Hyper-Link)與其他網頁建立關聯性，而這個關聯是屬於單向(1-Way)的，所連結的網頁也是以整個網頁內容為主[10]；Nelson T.H.提出一個將文件內容分割為許多部分，再進行內容重組的概念，如圖 2-2 與圖 2-3 所示，將電影片段剪接的概念運用於文件版本的變更，而版本間相同內容的段落產生關聯性，使用者可以利用此關聯進行版本之間的比較或相互參照(Cross-reference) [11]。

法國科學家 Julia Kristeva 曾提出「互為文本(inter-Textuality)」理論，認為世界上沒有一段文字是獨立存在的，任何一段文字一定會與其他文章中的文章產生某些關係，這關係就是加入標誌，電子文件在加入標誌後結構就與傳統的紙本文章不一樣了[12]。

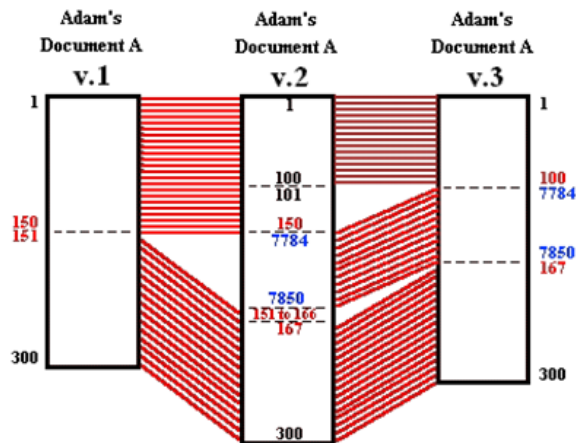


圖 2-2.不同版本文件內容示意圖[11]

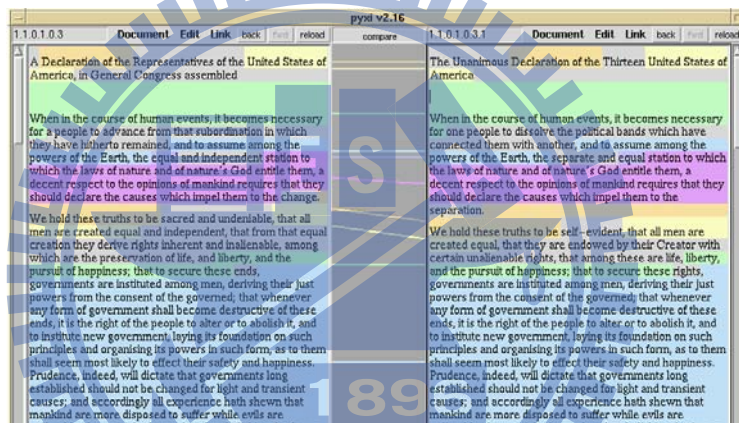


圖 2-3.不同版本文件內容比較圖[11]

而文件的內容不只可以從上一版本的內容中剪接至新版本，也可以將其他文件的內容段落直接引用至自身文件，並將引用的內容連結至原始文件，讓後來的讀者閱讀時可以更容易連結回原始文件進行參照閱讀。

另外 Nelson T.H.也提到內容連結(Content Link)的概念，它並非文件嵌入式的内容而是儲存於文件外部的參照內容，內容連結也可以建立兩個不同文件內容間的連結，如圖 2-4 所示，使用者 Barker 在他的文件中有一個區塊對 Adam 文件中某段內容進行了評論，則兩份文件的內容段落則產生了關聯性，使用者可由 Barker 的文件連結回 Adam 的文件閱讀原文或由 Adam 的文件參照至 Barker 的文件，讓文件間產生雙向的連結關係。

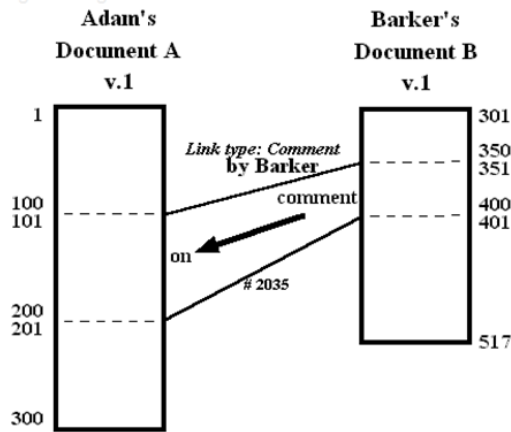


圖 2-4.內容連結示意圖[11]

2.1.3. 內容標誌

文章數位化後信息的表達產生改變，如表 2-1 所示，變成以自然語言與後設語言(Meta-Language)相輔相成的雙重結構，現在後設資料(Metadata)的部份已經有相當多的討論，像在圖書館的書目資料就是透過後設語言進行編目，而這些資訊都是來描述資料的背景資訊，如作者、年代…等，是適合敘述某類事物的共同現象(共相)，而無法顧及個別現象(別相)[13]。內容標誌所著重的就是內容本體，主要工作就是在處理個別內容描述的部分。

表 2-1.文章數位化後的訊息表達[13]

數位化之文章		表現系統
文章本身		自然語言
文章與外界的關係	情境描述(metadata)	後設語言
	參照聯繫(hyperlinks)	
	內容標誌(Content Markup)	

文字是作者表達文件抽象內容的媒介，而文件只是將概念表達在媒體上的「形式」(Form)[14]，文章經數位化後如不能將其著作時背景情境資料儲存起來，並與內容做適當的連接，就無法構成一個整體，情境對文章而言可泛指文章著作時的背景；而這些情境在作者完成著作時就已經確定，而這些背景訊息都是透過後設資料進行描述，如作者是誰、著作年代、作者生平…等，每一個領域的後設

資料定義也不盡相同；但真正對於我們人類要表達知識就一定要處理到內容的部分，而不只是處理背景資料。在內容標誌的部分，我們最常見的就是文章的篇章、節、段等結構上的標誌，也可以做一篇文章與另一篇文章的標誌；內容標誌主要的工作就是在處理文物的個別描述，以文章而言，就是對文章內容的理解、感受、比較與詮釋的部分。

文章的意義有兩層，一為作者原意，另一個就是讀者所理解到的意思，兩者經常並不完全相同，而且可能相差很遠[15]，所以也就有針對某篇文章或句子有不同的詮釋版本，像成語的部分，常會隨著風俗習慣或環境產生演變，如「夢寐以求」，出處為詩經<周南、關雎>，原意是指君子追求女子之心十分強烈，無論醒時或夢時都難以忘懷，後來這個成語就演變成不侷限於女子，而是泛指各種事物了。

在語義的部分，當一種形式可以對映到許多意義時，如何做正確的選擇，也就是所謂的義隨境轉的意思。就古文來說，要弄懂首先就要將文字的形、音、義標示清楚，然後了解其語法及詞性，字音的部分，中國字讀音約有一千四百種，常出現同音異義，因此讀音的標示就會影響到字義與詞義，所以字音與字義的連結性是相當強烈的。

在字義方面，中國字多有一字多義的情形，在不同的文章意境下表示不同的涵意，如果在閱讀的過程中沒有標示清楚，就會容易曲解作者的原意；此外在詞性與語法的標示，詞性常與字音字義產生關係，當動詞或名詞時的讀音是不相同的，詞性的標示也可以方便讀者研究古文對仗關係；古文在語法上也經常使用倒裝句或特殊語法，經由適當的標示也可以讓讀者更了解作者含意；而內容標誌最想做的是，就是與人合作，建立人機介面來做「了解」與處理「意義」的事，而這些事需要真正了解內容的專業人士來處理。內容標誌無論作理解、感受、比較、批評或詮釋，均觸及人文方面最根本的問題-意義(meaning)和了解(Understanding)[16]。

標點符號就是很典型的標誌，它改變了文章內容的表現方式，也是一種處理語意的工具，用的不同語意就不同，同一字串有不同的解讀[16]，例如「下雨天留客天天留我不留」，撰寫者的原意是「下雨天、留客天；天留，我不留」，但

閱讀者的解讀卻是「下雨天，留客天；天留我不?留」，標點符號使用方式不同，就造成同一字串的解讀不同。另外古文的句讀也是內容標誌之一，蘇東坡與曾國藩做過句讀標點的書是不同的，裡面還包含了兩位對內容的理解、詮釋的資料。所以標點符號與句讀這類標誌，都是設計來協助讀者理解文章內容的[16]。

2.1.4. 中文文獻系統

中央研究院資訊科學研究所文獻處理研究室為探討古籍多版本表達方式與版本間的相互比較與轉換，開發了「中文文獻處理系統(Chinese Document Process,CDP)」，如圖 2-5[17]，陳昭珍也利用此系統來解決古籍各類資訊之多版本導行問題；此外也被利用來研究佛經的多版本[18]。

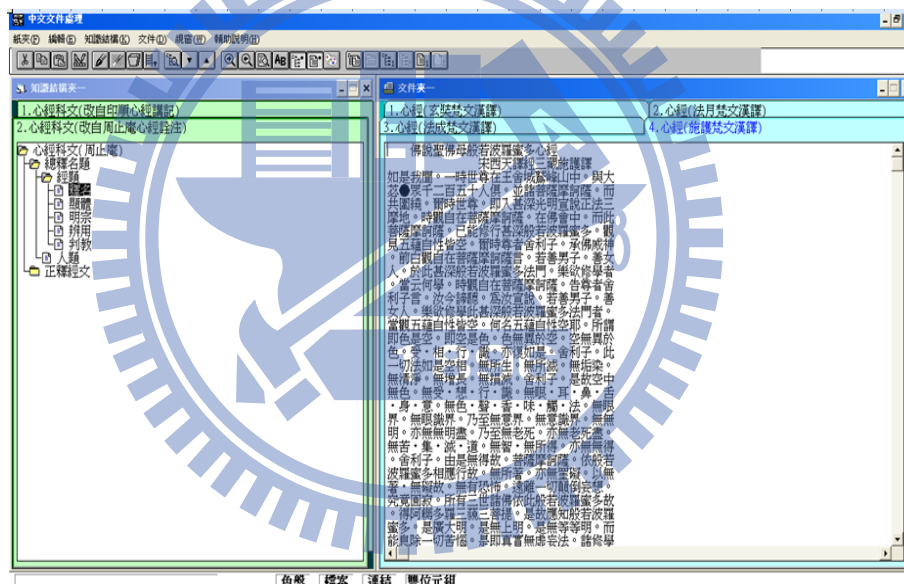


圖 2-5. 中文文獻系統

通類知識是 CDP 系統用來處理文獻的基本概念，為一樹狀結構，每一分枝就是超文件的節點[20]，相關研究者分析古籍文件所擁有的通類知識，以佛經而言，利用了科文來建立通類知識；Roy Radal 認為對作者而言，創作的過程分為探究(Exploring)，組織(organizing)，撰寫(Encoding)三階段；而對讀者而言，對文件的解讀則是反方向的[20]。

中央研究院以心經多版本為例，將心經科文視為系統的通類知識，利用科文

來建立心經不同版本間的對映；陳昭珍[5]利用中文文獻系統進行文心雕龍進行文件多版本的超文件的探討，文獻中提到研究者在研究文獻時，需要正文、傳注與版本資訊來協助研究；而這些資訊均具有多版本的性質，研究者如何透過系統來協助研究，就是一個重要的課題，她就利用中文文獻系統來解決多版本文件間導行的問題。首先利用知識樹功能建立通類知識，研究中將四庫分類結構表、古籍傳注體系表…等結果建置成系統通類知識。

王梅玲[3]研究的實驗流程中，先透過對文章內容的了解後建立通類知識樹狀結構，才將文件轉入中文文獻系統，將樹狀結構分為章法分析、修辭格…等結構，再將通類知識與文件利用系統建立成一關係圖，如圖 2-6 所示。

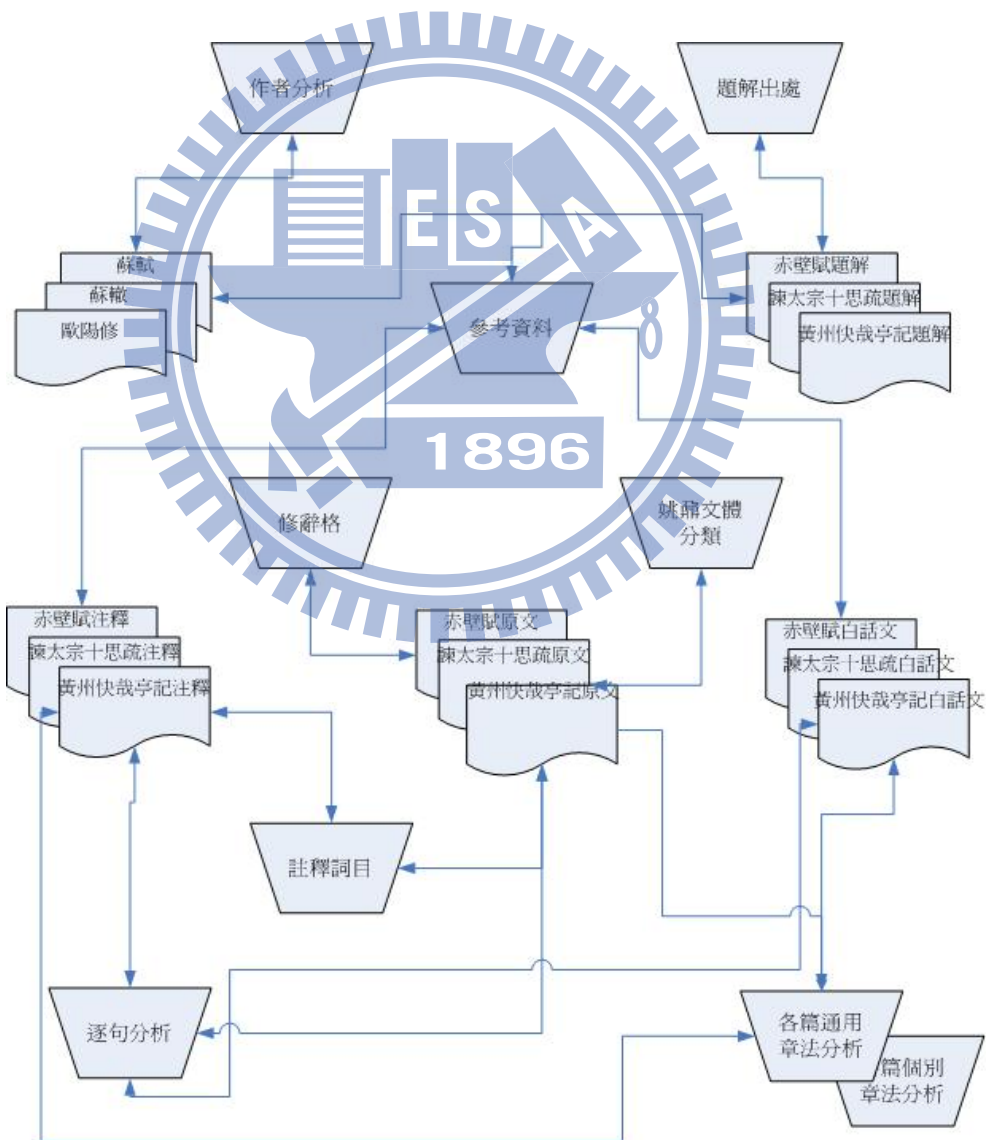


圖 2-6.知識結構與文件關聯圖[3]

王梅玲[3]也提到了實驗中發現了系統仍待解決之問題，樹狀結構的節點與文件內容的連結關係，有一對一、多對一及一對多等三種可能，也就是說，一個節點對應一段文字內容、多個節點對應於同一段內容或是一個節點同時對應好幾段內容。目前系統可做到前二者連結關係，而無法同時對應幾段內容。這使得在一些連結上產生了問題。例如：做修辭的標誌連結時，當同一篇文章中有多處是以同一種修辭方法做描寫時，由於目前系統尚未提供一個樹狀結構的節點，可以對應多個標誌的內容之功能，所以並不能一次將所有的詞句對應出來。目前有一變通方法，就是在修辭格結構之下再加一層，讓各種修辭格成為多個節點，例如：擬人法之下，加上擬人法一、擬人法二...，再讓運用同一修辭法的文句對應到各節點上。但是當資料量不斷增加後，這並不是很好的解決方式。

2.1.4. 文件內容注釋

1. 文件注釋的好處

以往在閱讀文件資料時，常常會在閱讀的過程中記錄當時的想法與內容的重點，我們最常使用的方式為利用筆記本寫下註記[20]，或直接將註解或想法直接註記在紙本上，以利再閱讀時可以更快的了解內容的重點，忽略掉不重要的部分以縮短再閱讀的時程，這也是為什麼使用者喜歡購買或閱讀他人所註記過的文件；如果是線上文件，也只能利用筆記本或電腦文件處理軟體(Word Processing)來記錄[21]，但缺點在於在往後的閱讀還需要將註記內容與文件本身做對映才能發揮效果[22]。

Kawase 透過實驗，發現使用者在紙本中注釋的習慣分為：在標題畫底線(Underlining)或註記(Highlighting)、在內文中的文字做註記、在圖形周圍或其他空白處寫下想法或評論[23]。而使用者在註記時，會用不同的形狀或顏色在區分內容的重要性，如圖 2-7 所示。

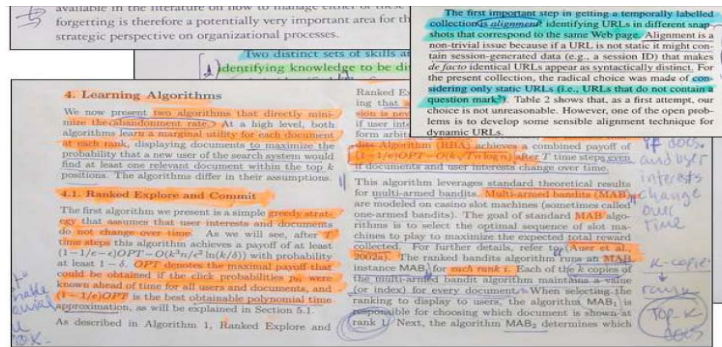


圖 2-7.傳統注釋內容[23]

文件注釋對於閱讀者的好處則包含資訊再尋找(Refinding)的方便性，協助了解、記憶及未來閱讀時讓閱讀者容易注意[23]，這些好處不只侷限於自己，也能延伸至往後其他的閱讀者[24]。

線上文件注釋除了承襲了傳統紙本(Paper-based)注釋的好處外，使用者還不受時間與空間的限制，與其他使用者分享與討論，達到合作學習的效果[25]，因為通常在閱讀文章遇到問題時，只能將問題暫時記錄，再與他人進行討論或解答，線上文件注釋系統就能達到更即時的效果，對於使用者學習新知識也更有幫助[26]。

文獻[27]利用實驗來比較搜尋引擎(Search Engine)、電子書籤(Bookmarking)與注釋工具(Annotation)間資訊再尋找的時間，發現使用者使用注釋工具能最快的找到日前所留下的有用資訊，如圖 2-8 所示。

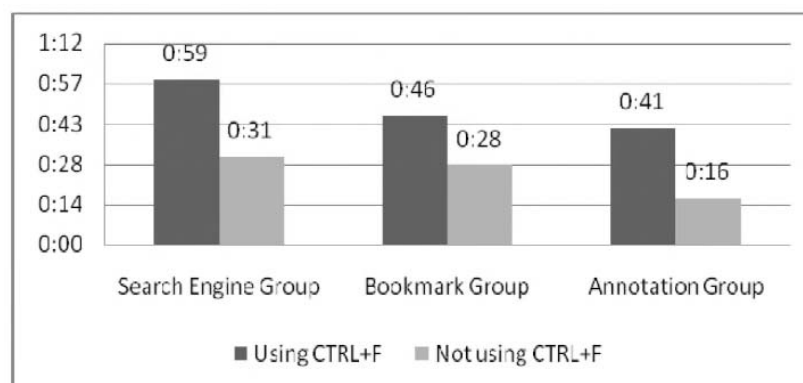


圖 2-8.資訊找尋時間比較圖[27]

2. 文件注釋工具調查

Hoff C.[24]將電子文件注釋系統進行收集與分類，以圖 2-9 所示，將系統依格式、注釋方式、管理與實作進行分類，而格式細分為文件格式與注釋格式，而注釋的方式也分為三類：In-Place 為在段落與段落間插入注釋內容的區段；On-Place 則直接緊鄰顯示於被注釋文字旁，此方式在傳統注釋方式是無法達到的；Off-Place 將所有注釋顯示於本文之外，如圖 2-10 所示；而分享功能則在管理互動中進行區分。

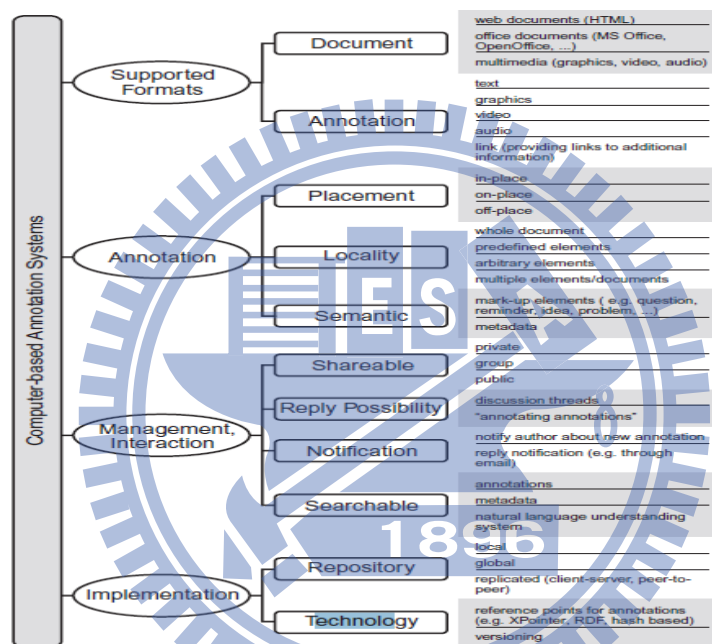


圖 2-9. 注釋系統分類圖[24]

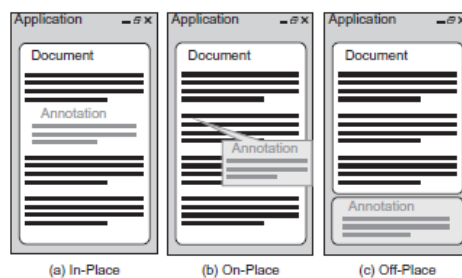


圖 2-10. 注釋方式表示圖[24]

文件注釋不僅能幫助使用者了解文件的重點，透過分享的功能更能促進使用者間的討論，而電子文件注釋功能可以讓使用者透過遠端存取而達到分享的目的，以下為常見的電子文件注釋功能[28]：

- 連結注釋(Link Annotation)：使用者透過連結注釋來導航與搜尋不同的文件，而連結可分為粗略(coarse-grained)或細微(fine-grained)，或者可以單向(uni-directional)或雙向(bi-directional)來區分。
- 名稱與位置注釋(Naming and Addressing Annotation)：註記(Notes)與連結(Link)能被導航(Navigation)與搜尋(Search)，而位址(Address)可以是網路上的全球資源定位器(URL,Universal Resource Locator)。
- 索引(Indexing)：將文件中的注釋以列表(List)來表示，此功能提供文件內容概述，也方便於編輯、排序與管理。

目前線上文件常以 PDF(Portable Document Format)文件檔案為主要發行格式，PDF 的主要注釋模式有三種：[29]

- 加注解釋在整份文件上，但無法知道註解是屬於文件中的那一段，註解與文件本身是分開的。
- 註解可以針對文件中的某個段落或部分下註解。而可以由特定的段落找到相關的註解。所加入的註解是跟隨著文件(放在文件中的metadata)
- metadata 與文件是分離的，文件如果被複製或移動，metadata 是沒有辦法跟隨的。

在文獻[29]中所提到的系統PDFTab主要是利用第二種模式將文件(Document)與知識本體(Ontology)整合，將Metadata放置於文件中，PDFTab利用OWL(Web Ontology Language)來儲存Ontology，因為PDFTab 建構於Protégé系統基礎上，而Protégé容許使用者以OWL 的方式對PDF 文件做注釋，而PDF 本身所提供的RDF與OWL並不能共通使用。

Adobe公司所開發的Acrobat就有提供PDF內容注釋工具，但注釋工具需視文件的安全性是否開發而決定能否使用，而該工具也無法進行分享的動作，產生不

少的限制；而在網頁注釋的工具方面，近年來也是有所發展，表2-2為目前市場上網頁注釋工具的比較表。

表2-2.網頁注釋工具比較表

function name	Diigo	Stickis	ShareCopy	MyStickies	WebNotes	YAWAS
Highlight	√	√	√	√	√	√
Add Notes	√	√	√	√	√	
Tag (categorize)	√	√				
ShareNotes	√	√	√		√	
Group	√	√				
Search	√			√	√	

資料來源：本研究

2.2. 使用的程式工具

系統所提供的文件定位需要進行換算，避免造成伺服器端運算負荷過大，本系統將定位計算與介面服務交由用戶端來執行，所以使用 Javascript 來協助開發用戶端程式，並透過 ExtJS 3.0 網頁套件來設計介面，以節省介面設定的時程；在伺服器端則利用 Python 進行資料庫存取；系統也利用 Graphviz 來將使用者所建立的資料來顯示文件、節點與使用者之間的關聯圖，讓物件之間的關係可以清楚顯示。

2.2.1. Javascript

Javascript 是早期由 Netscape Communications 公司所開發的網路描述語言 (Script Language)，前身是 LiveScript，與 Sun Microsystems 公司的 Java 語言是不同的，兩者的差異如表 2-3；Javascript 專門用於網頁中，且是執行於用戶端，以提高網頁互動性為目標，主要功能如下[36]：

1. 提高網頁互動性及趣味性。
2. 計算與驗證執行於用戶端，減少伺服器的計算與流量。

3. 可利用 Cache 功能將未用到的圖檔預先抓回，增加便利性。
4. 可在不換頁的情況下來存取伺服器資料並顯示於網頁。

表 2-3.JavaScript 與 Java Applets 比較表[37]

JavaScript	Java Applets
由客戶端的 JavaScript 解譯器進行逐列解譯後執行。	由伺服器取得編譯後的 Bytecode，然後在客戶端由 Java Virtual Machine 執行。
物件基礎 (Object-based) 的語言，繼承 (Inheritance) 關係必須經由特殊方式才能達成，性質及方法可以動態地加到一個物件。	物件導向 (Object-oriented) 的語言，物件可分為類別 (Classes) 及實例 (Instances)，繼承關係來自於物件的階層性。類別及實例都無法具有動態產生的性質及方法。
程式碼內嵌於 HTML 網頁之中。	以特殊標籤來將 Java Applets 加入網頁之中。
所有變數不需要事先宣告資料型態，即可運行指定變數值。	所有變數都必須事先宣告資料型態。
在執行程式碼時，才會檢查所到的物件是否存在。	在編譯程式碼時，即會檢查所用的物件是否存在。
無法讀寫客戶端的硬碟 (Cookies 除外)。	無法讀寫客戶端的硬碟。

微軟在 Internet Explorer 3.0 版時支援 Javascript 1.0 版，稱為 Jscript，是一種與 Javascript 相容的 Script 語言，初期不是很穩定，且問題多，到了 Internet Explorer 4.0 版時支援官方 ECMA 標準的 Javascript 規格，但與 Netscape 的 Javascript 是屬於兩種不同的 Script 語言；而 Javascript 的特點如下[36]：

- Javascript 不是 Java 程式語言，它屬於一種 Script 語言。

- Javascript 是一列一列執行的 Script 的程式碼。
- Javascript 的程式碼是直接嵌入 HTML 文件，屬於 HTML 文件的一部分。

2.2.2. ExtJS 3.0

ExtJS 是一種以 Javascript 基礎所開發的客戶端框架(Framework)，與後端技術無關，由 Sencha 公司所開發出來，主要開發者為 Jack Slocum，於 2007/4/1 發行了版本 1.0，；ExtJS 的前身是 YUI(Yahoo User Interface)的應用，後來獨立出來成為一套專案，其長處在於使用者介面(User Interface;UI)上，提供了許多精美的介面讓開發人員使用，例如：Panel、MessageBox，Form…等，只要透過 API(Application Programming Interface)使用 ExtJS 所提供的應用，設計出精美的程式介面；ExtJS 也開放了原始碼供開發人員修改，讓開發人員可以需求自訂不同的功能，可將 ExtJS 運用於 .Net、Java、PHP 等開發項目；ExtJS 至今仍然持續開發，在官方網頁也提供使用者詳細的使用方法與許多的範例，讓開發人員能輕易的了解與使用，其架構如表 2-3 與圖 2-11 所示[38]。

表 2-4.ExtJS 函式庫[38]

Base Library	Include Order
Ext Standalone	ext-base.js ext-all.js (or your choice of files)
Yahoo! UI	yui-utilities.js (or your choice of YUI base files) ext-yui-adapter.js ext-all.js (or your choice of files)
jQuery	jquery.js jquery-plugins.js // required jQuery plugins ext-jquery-adapter.js ext-all.js (or your choice of files)
Prototype / Scriptaculous	prototype.js scriptaculous.js?load=effects (or whatever you want to load) ext-prototype-adapter.js ext-all.js (or your choice of files)

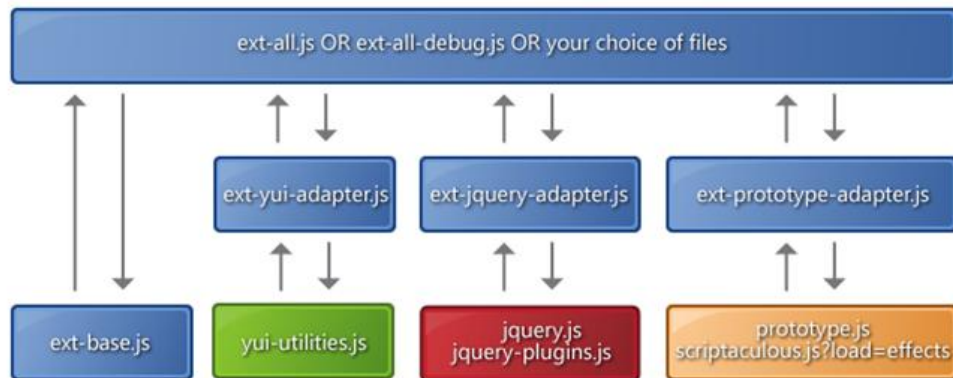


圖 2-11.ExtJS 架構圖[38]

ExtJS 所支援的主要網頁瀏覽器包含：

- Internet Explorer 6+
- FireFox 1.5+(PC,Mac)
- Safari 3+
- Chrome 3+
- Opera 9+(PC,Mac)

2.2.3. Python 2.5

本系統在接收到用戶端所回傳的資料，需要在伺服器上進行運算或處理，並透過伺服器與資料庫做資料存取，目前可以運用的程式語言有許多，例如：CGI、PHP、ASP...等，都是常見的網頁程式語言；本研究利用 Python 語言設定伺服器端程式，它的優勢在於語法格式清晰，強迫縮排的特性，讓程式碼的可讀性提高，有完整的物件導向，完全的跨平台，在網頁的開發也可與 Zope 或是 Plone 的架構來開發，還可方便與 C 或 Java 結合，可簡短開發時程[39]。

Python 的程式碼簡潔，提供大量的程式模組，往往只需數行程式就可以達到其他程式數十行的效果；Python 是一種直譯式電腦語言，由 Guido van Rossum 創造，在 1991 年 2 月正式發表，Python 的優點如下[40]：

1. 易讀性：語法簡潔易讀，讓使用者可以快速學會並建立高效率的程式碼。

2. 高支援性：程式碼是公開的，全世界有無數的人在協助搜尋 Bug 並修訂它，且不斷的加強功能，讓它成為更有效率的程式語言。
3. 快速建立程式碼：內建直譯器，可直接於直譯器內執行、測試與執行，不需額外的編譯器。
4. 重複使用性：以模組(Module)及類別庫(Package)來儲存，可在不同的程式中使用。
5. 高移植性：多數的模組可在多種類的作業系統中執行，只有少部分針對特殊作業系統而設計。

2.2.4. Graphviz 圖形表示工具

現在許多軟體工程常利用圖形化來表示結構且抽象化的概念，讓使用者對於內容結構能一目了然；自動化圖形產生工具在軟體工具、網站設計與資料庫設計就扮演了非常重要的角色，讓繁複的系統關係能用簡易的關係圖讓使用者可以快速瞭解。Graphviz 視覺化圖形工具提供了多種圖形種類，是一種由 AT&T 所開的圖形產生器，屬於開放式原碼(Open Source)程式，讓使用者可以針對圖形內容進行編修[42]，更透過命令程式(Command)讓其他程式進行呼叫，自動依指令產生圖形，Graphviz 的特點就是只是提供節點(Node)之間的關係，系統就可透過工具演算法繪出系統關聯圖來表達節點的特性與節點之間的關係。目前工具所能提供的圖形種類有 dot、Neato、fdp、sfdp、twopi 與 circo 來顯示不同類型的結構。

第三章、 系統分析與設計

3.1. 系統設計概念

本系統所設計的概念是基於知識共享的理念，以文件為基底，讓使用者可以將相關資訊堆疊於文件之上，讓使用者間可以相互分享，此概念與 Wikipedia 使用者可共同編輯內容的概念相似，但本系統並不變更文件內容本身，而是將額外訊息與文件產生連結，避免原有內容失真。

此外，系統也透過知識結構，讓文件間可透過知識節點產生關聯，如圖 3-1，讓使用者可以透過節點即能找尋相關文件之間，讓使用者可以短時間找到相關文件或比較文件間的異同。

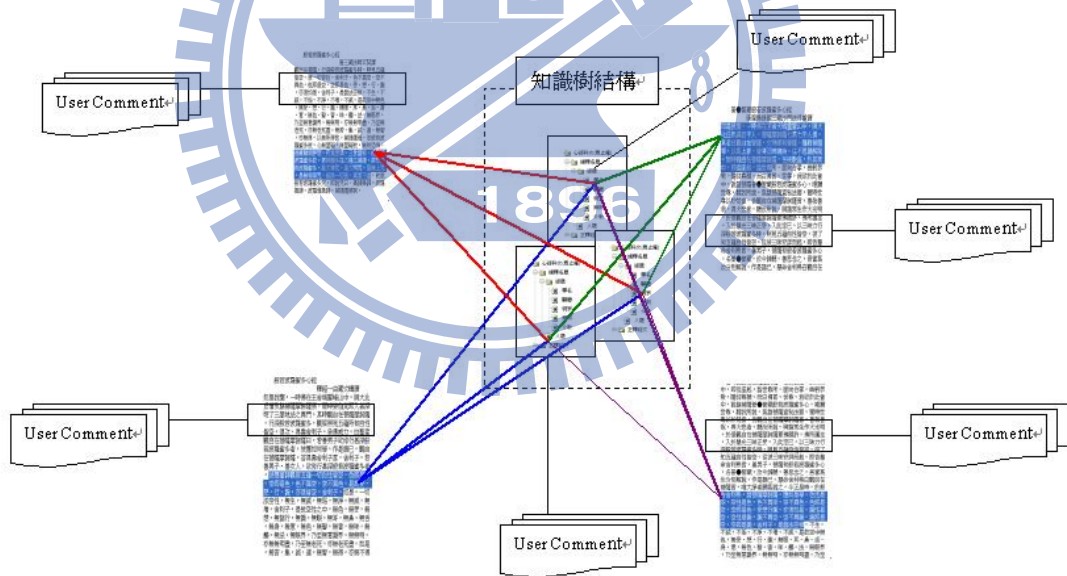


圖 3-1.系統概念圖

3.2. 設計架構與模式

本系統以三層式(展式層、商業邏輯層、資料存取層)架構來建立，如圖 3-2，系統將 MVC 設計模式中的 Model、View 及 Controller 分散在客戶端與伺服器端，以減少伺服器的負荷。

1. 三層式架構

- 展示層(Presentation Layer,PL)

客戶端透過網頁瀏覽器(如：Internet Explorer、Firefox)與伺服器互動，本系統在此層不僅是進行資料展示，也進行資料的運算，將運算的結果再回傳給伺服器，避免因所有運算均在伺服器而造成伺服器負載過重。

- 商業邏輯層(Business Logic Layer,BLL)

此層接受客戶端的請求，處理及運算使用者所提供的資料及傳送使用者所要求的資料，可視為是使用者與系統資料庫間的橋樑。

- 資料存取層(Data Access Layer,DAL)

主要提供資料給商業邏輯層；因客戶端無法直接存取資料庫，必須透過系統程式的控制，以提高系統安全性。

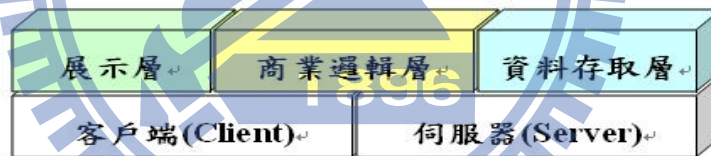


圖 3-2.三層式架構圖

2. MVC 設計模式

MVC 是軟體工程的一種軟體架構模式，將軟體分為模型(Model)，檢視(View)及控制器(Controller)，如圖 3-3，最早是由 Trygve Reenskaug 於 1974 年提出，MVC 的目的是實作一種動態程式設計，讓後續對程式的修改和擴充簡化，讓一部分程式可以重複利用，也讓程式結構更加直覺。

- 模型元件

程式設計師撰寫應有的功能或演算法，也可以針對資料庫進行存取與管理的動作。

- 檢視元件
主要在介面設計的部分，將模型元件的資料展示給使用者。
- 控制器元件
負責使用者請求(Request)的處理，協調模型元件與檢視元件。

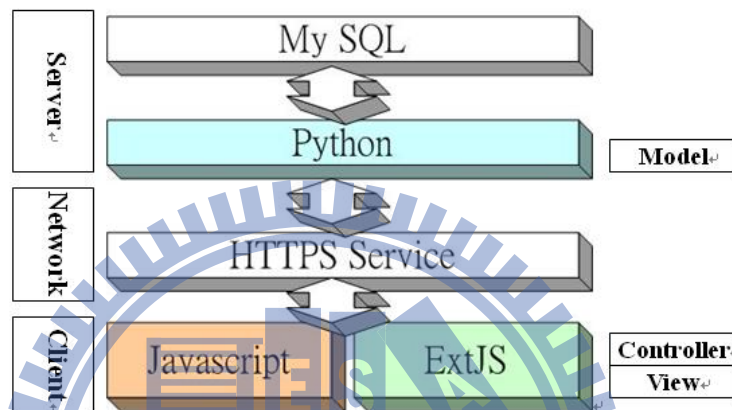


圖 3-3. 系統 MVC 設計架構圖

3.3. 系統模組設計

圖 3-4 為注釋系統設計架構，系統主要分為五個模組分別為：

1. 文件工具模組：使用者可透過此模組進行上傳動作，上傳的文件類別可為純文字(Text)檔案或 PDF 檔案類型，系統將檔案轉換為網頁(HTML)型態顯示；使用者也可透過此模組載入檔案或將文件加入個人收藏。
2. 知識架構模組：使用者可透過知識架構模組進行知識樹內容的編輯；使用者也可將其他使用者所分享的知識結構加入我的收藏。
3. 內容對映模組：使用者利用此模組進行知識樹節點與內容段落的對映；使用者也可選取文件內容的段落來尋找所對映的知識樹節點，或反向選取知識樹節點來找尋所對映的文件內容段落。

4. 內容注釋模組：使用者可透過此模組選取知識樹節點或文件內容任一段落進行注釋，並分享給其他使用者。
5. 個人收藏模組：使用者透過此模組管理個人收藏，選擇是否分享所建立的內容與管理本身所加入的文件。

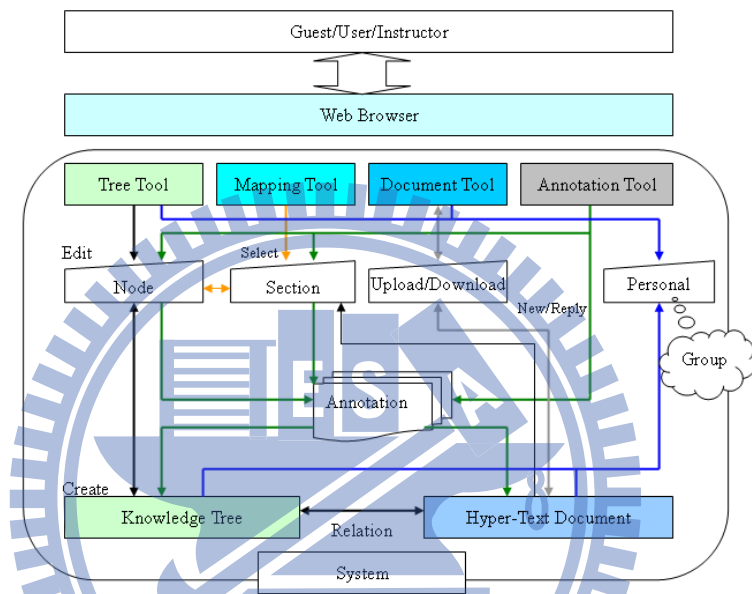


圖 3-4.系統設計架構圖

3.3.1. 文件工具模組

此模組分為三個功能，如圖：

1. 文件上傳功能：使用者上傳純文字(Text)或 PDF 檔案，透過系統轉換功能轉換為網頁檔(HTML)，在轉換的部分，系統直接將純文字檔案內容處理加工後另存成網頁格式；而 PDF 檔透過 VeryPDF PDF2Word 3.0 轉換程式，先將 PDF 檔案轉換成 Word 格式後，再利用 Word2Html 程式將 Word 內容轉換成網頁格式後再系統會使用的程式嵌入網頁中，然後將檔案存入文件庫與資料庫後方可使用。
2. 文件載入功能：使用者選擇類別後，可載入該類別文件列表。

3. 個人收藏功能：將有興趣的文件加入個人收藏庫中。

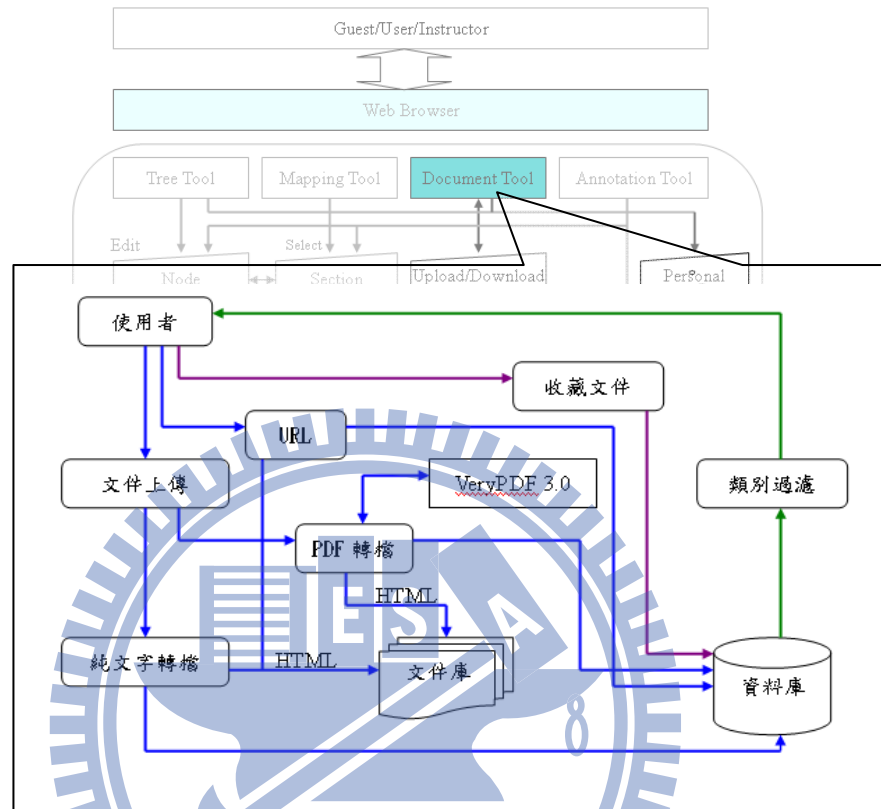


圖 3-5.文件工具模組流程圖

3.3.2. 知識架構處理模組

此模組包含知識樹與節點的新增刪除功能，此模組需進行身份判斷，每個知識樹均屬於唯一使用者所有，他人無法編輯知識樹內容；使用者也可將有興趣的知識樹加入個人收藏，如圖 3-6。

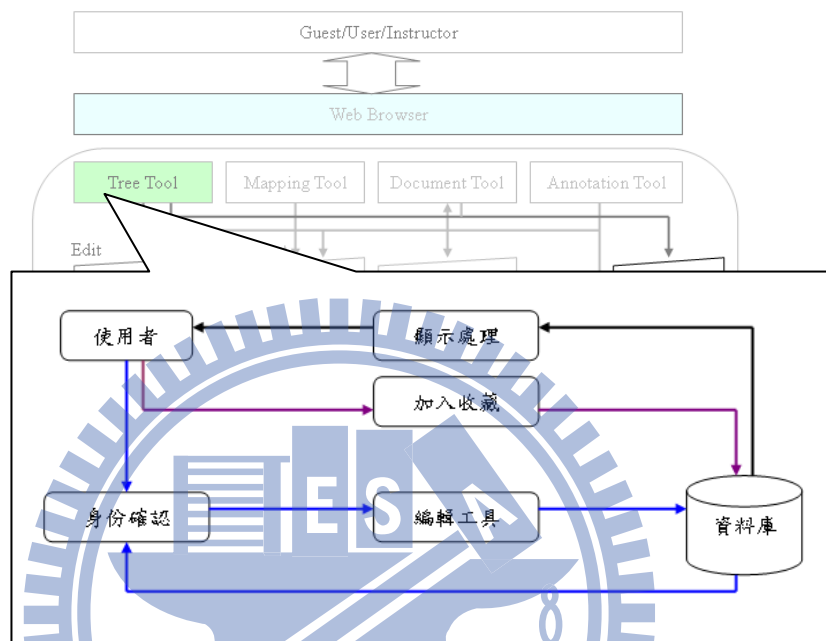


圖 3-6.知識架構處理流程圖

3.3.3. 內容對映模組

此模組含有知識樹節點與文件內容的對映，如圖 3-7；當使用者選取節點及文件內容段落時，經有系統對映模組中的字元計算，取得選取文字位置後，將內容存入資料庫，建立關聯；反之也可由節點尋找所對映的文件段落或由使用者所選取的文件段落來尋找此段落是屬於那一個節點。

系統中文字定位演算法是透過 Javascript 中 CreateTextRange 的特性所編寫，如圖 3-8；Javascript 的 TextRange 函數符合系統讓文字產生被選取的狀態；本研究的計算方式以 TextRange 中的 Move 方法為中心，用此方法來計算出使用者所選取的開始位置與所選取的文字長度；因 Move 方法中可以使用字元 (Character)、單字(Word)與段落(Sentence)來進行移動選取。

如果直接從頁面文字起始至所選取的範圍的長度來計算選取的起始位置，在使用 TextRange 函數來顯示所選取的文字時，將會產生誤差，因為 String.Length 函數並不會將字串中的特殊字元加以計算，所以如果頁面內容如果含有特殊字元，就會在顯示時產生誤差；因系統以 TextRange 的 Move 方法為選取範圍顯示函數，所以系統計算方式以模擬使用者選取文字的方法來與真正使用者所選取的內容做比較，避免所得到的起始位置與文字選取長度產生誤差。

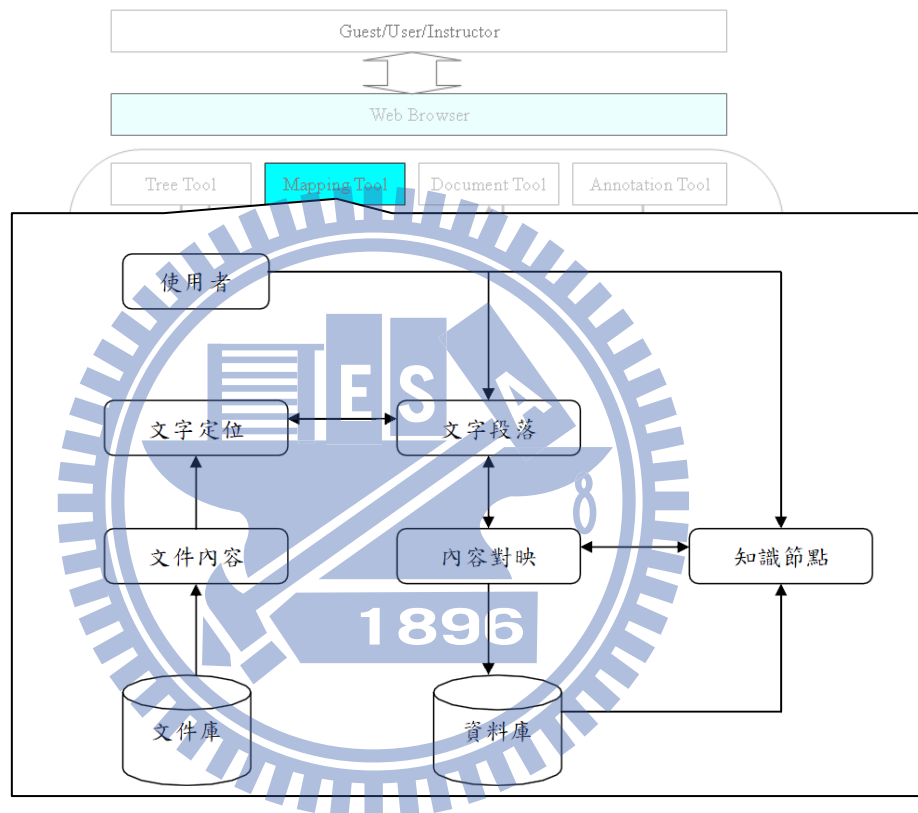


圖 3-7.內容對映模組流程圖

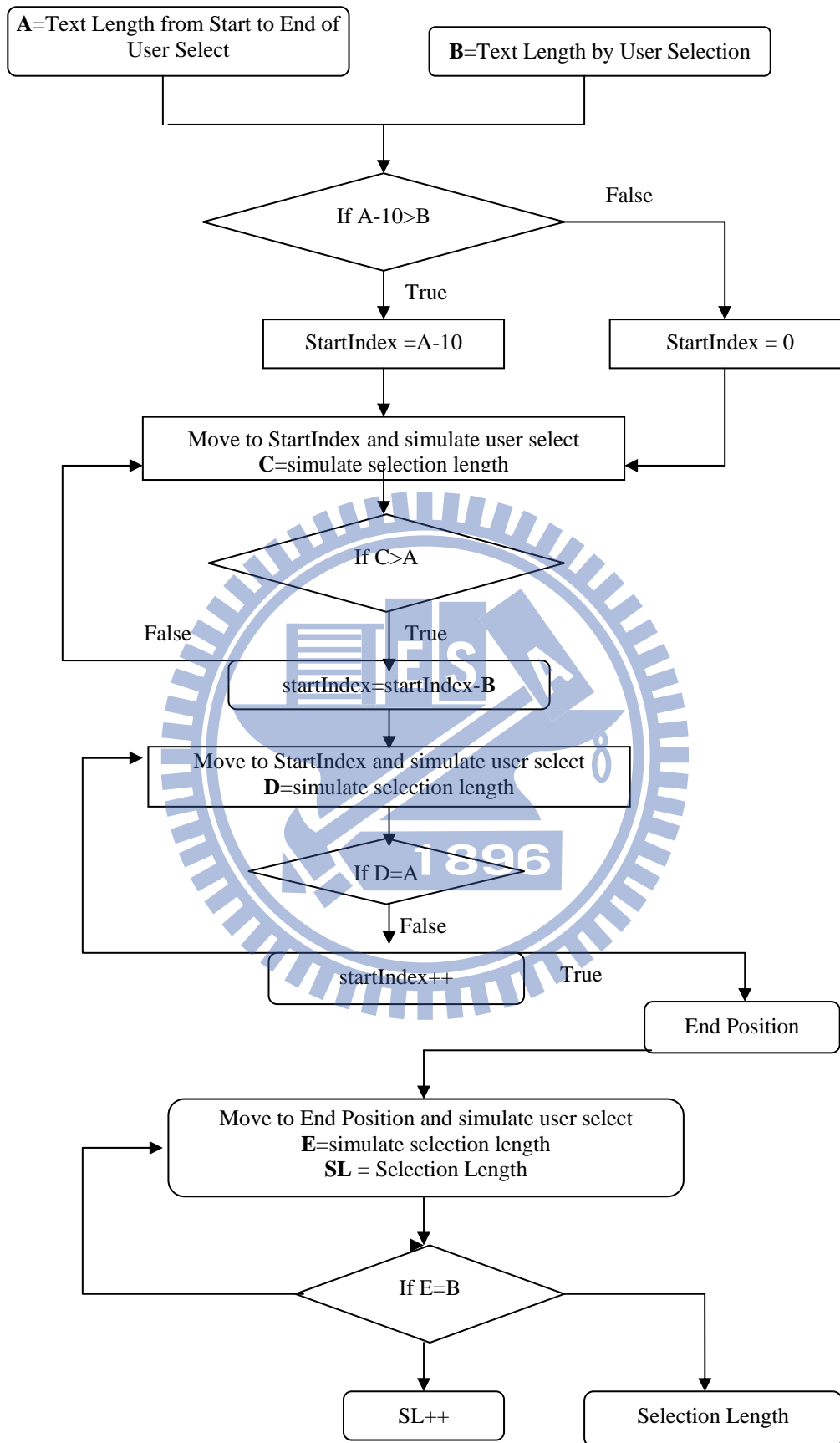


圖 3-8.網頁文字定位流程圖

3.3.4. 注釋模組

此模組可針對知識樹節點與文件內容進行注釋，使用者選取文件段落後，經利用 3.3.3 節中演算法計算文字選取位置後，系統顯示關鍵字及文字位置後，再將使用者所輸入的註解存入資料庫中，其他使用者也可針對相同的內容進行加註動作；另外使用者也可以選取知識樹節點進行注釋動作，如圖 3-9。

在文字段落注釋部分，使用者選取的內容中產生重疊的部分，系統對於此情形將以大範圍的段落為主，但在注釋內容中以關鍵字來區分，避免使用者產生困惑；如下二列文字，使用者分別選取一時佛…薩眾俱，與王舍城…山中，系統在顯示時只會表示出一時佛…薩眾俱的段落而不會特別在段落中註明王舍城…山中也有被選取。

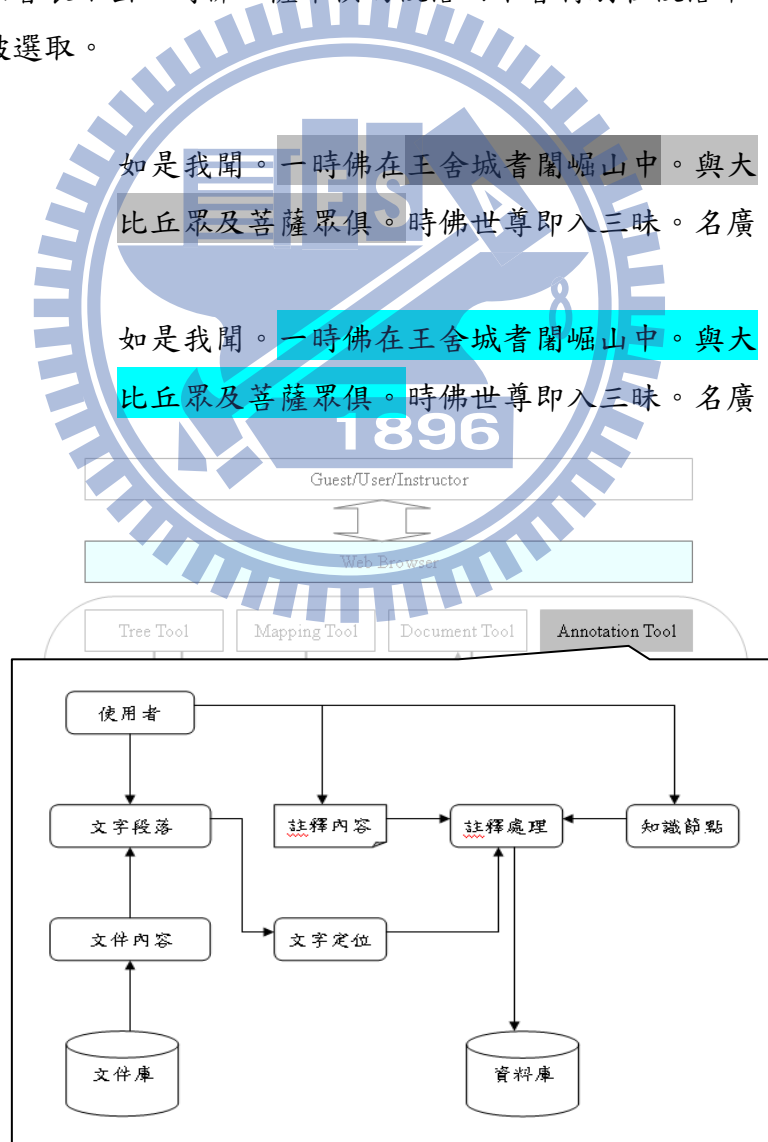


圖 3-9.注釋模組流程圖

3.3.5. 個人收藏模組

當使用者透過文件及知識樹工具將有興趣的資料加入我的收藏後，可透過個人收藏模組管理我的收藏，如圖 3-10；如果該文件或知識樹剛被建立時，系統預設資料是未公開的狀態，資料擁有者需透過此模組進行資料公開的操作才可分享給其他使用者；如果使用者加入的資料為其他人所建立的，可以進行刪除管理，如果是個人擁有且已公開的文件，系統就不給予刪除管理，避免文件已有建立相關資訊或連結，造成文件刪除後相關資訊也一併遺失。

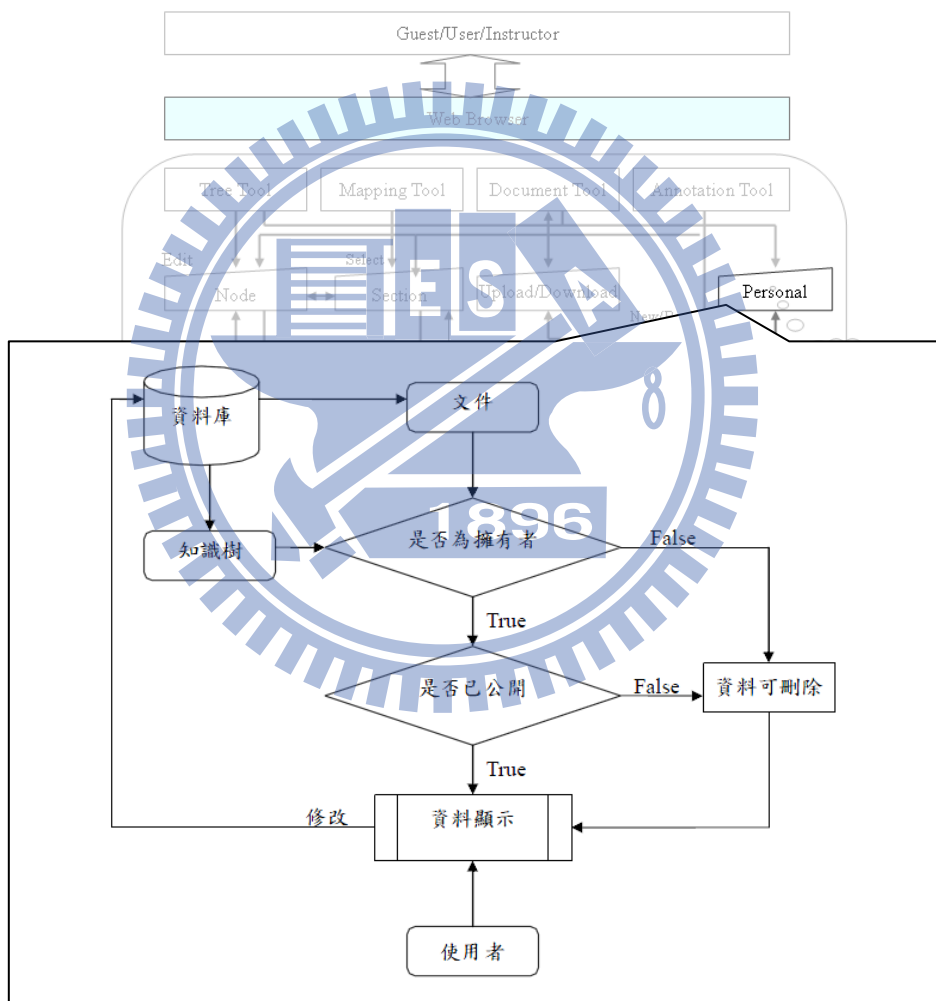


圖 3-10.個人收藏模組流程圖

第四章、系統功能展示

本章內容分為兩個部分，4.1 節介紹系統實作的環境；4.2 節介紹系統實作結果。

4.1. 系統實作環境

本研究的實作環境如圖 4-1 所示，系統作業平台為 Window Server 2003，使用 XAMPP Win32 1.7 做為伺服器工具，內含 Apache Httpd(網頁伺服器)、MySQL(資料庫)…等網站所需工具，本研究所使用的 Python CGI 網頁也是透過 Apache 引擎進行處理，並利用 XAMPP 中所提供的 MySQL 為本研究之資料庫系統，系統則利用 Python、Javascript 及 ExtJS 為開發工具。

注釋系統利用 XAMPP 將所有網站伺服器需要的套件整合的便利性，而且沒有屬於自由軟體，沒有版權問題，而 XAMPP 中的 Apache 網頁伺服器可以處理許多類型的網頁，如 PHP、CGI、Python…等，讓開發過程更為便利順暢，其中更整合 MySQL 資料庫，讓網頁及資料庫的連結在安裝後即可使用，讓網頁與資料庫間的參數設定，節省開發時程。

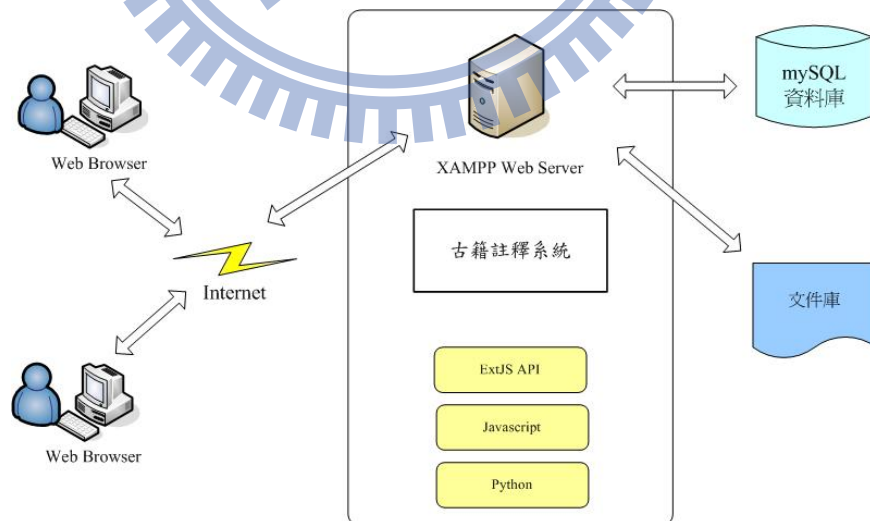


圖 4-1.系統實作環境圖

4.2. 系統實作結果

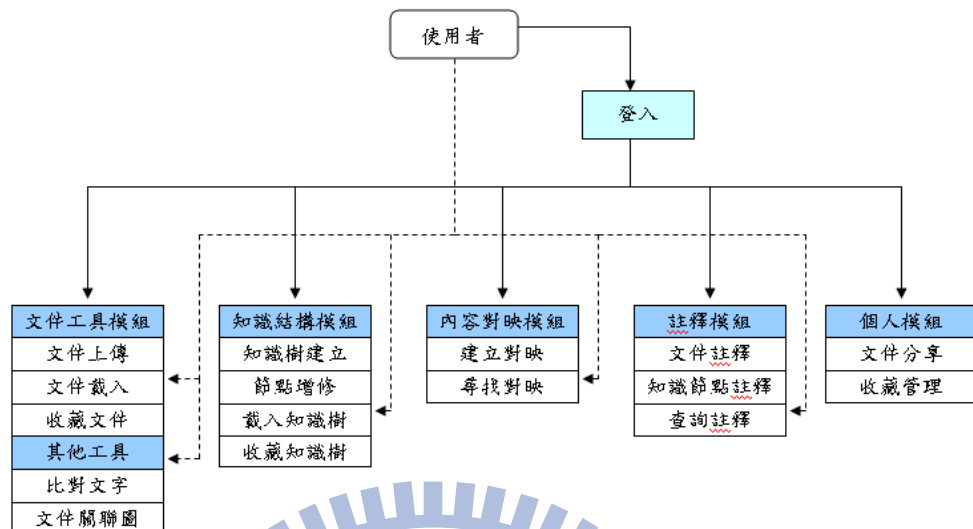


圖 4-2.系統功能架構圖

注釋系統依前面章節所描述的系統設計，注釋系統主要分為：文件及知識樹功能、對映功能、注釋功能及個人收藏功能等五大部份，如圖 4-2；系統中有許多功能需記錄使用者相關訊息，所以必須經過身份認證才可以使用，所以使用者需要登入系統才能使用所有功能，未登入使用者只能使用圖 4-2 中虛線所表示的功能。

4.2.1. 文件功能

1. 文件上傳

使用者在登入系統後，可以利用「文件上傳」功能上傳純文字(Text)檔案或 PDF 檔案，如圖 4-3，上傳後系統會經過程式轉換成網頁(HTML)；如果上傳的為純文字檔案，系統只將內容讀入後加入網頁 HTML 跳行編碼
，並嵌入 Javascript 程式碼(/js/doc.js)後存入 html 資料夾並將相關資訊寫入資料庫後回覆轉換結果給予使用者，如圖 4-4；PDF 檔案格式則由需求 VeryPDF PDF2Word 3.0 程式先行轉成 Microsoft Office Word 後利用自行開發的 Word2html.exe 程式嵌入 Javascript 程式碼來進行格式轉換。



圖 4-3.文件上傳畫面



圖 4-4.文件上傳結果

2. 文件載入

「文件載入」功能不需身份認證，一般使用者就能開啟，在選擇類別後，開啟文件列表，系統會將自己所上傳(需登入)或他人所分享的文件列出，讓使用者載入系統中，如圖 4-5；選擇文件後，就可以於文件內容瀏覽區塊開始閱讀文件，如圖 4-6。而在文件載入後，系統在類別關聯名單區塊會顯示有將使用者所選類別中的知識結構或文件加入個人收藏的使用者，而在文件關聯名單區塊則顯示有對此文件下過注釋的使用者，利用此功能建立簡易社群關係，如圖 4-7 所示。



圖 4-5.文件載入功能



圖 4-6.文件閱讀區塊

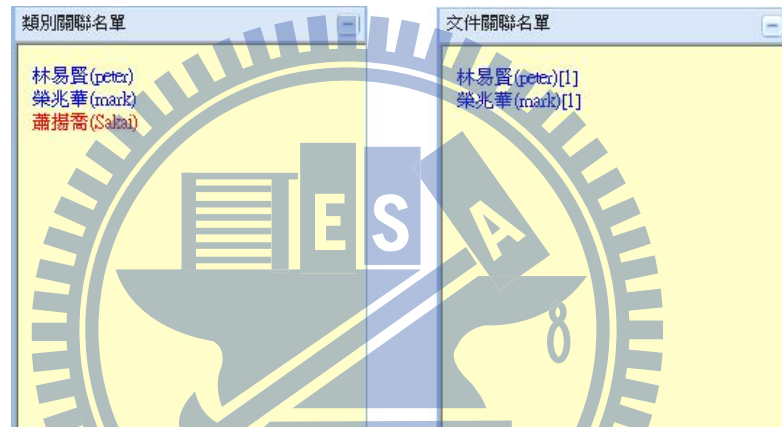


圖 4-7.關聯名單顯示

4.2.2. 知識結構功能

1. 知識樹管理

使用者在登入後，須選擇類別後新增知識樹，新增後知識樹會與使用者產生關聯；在知識樹列表中，所載入此類別中個人所建立或他人所分享的知識樹內容；在刪除的部份，只能透過個人收藏進行刪除未分享的知識樹內容。新增畫面如圖 4-8。



圖 4-8.知識樹新增畫面

2. 知識樹節點管理

使用者選擇知識樹節點後，知識樹的擁有者可以進行節點的新增與刪除，節點刪除後，關於節點的相關訊息將會一併刪除；如圖 4-9。



圖 4-9.知識樹節點新增畫面

3. 知識樹節點的應用

王梅玲提到文章的寫作通常由作者由簡單的想法演譯成為一篇文章，而讀者閱讀文章時則透過歸納的方法來擷取文章重點，或者將文章內容依領域的不同加以分類[14]，如詩品寫作風格可依二十四詩品或賦、比、興等來區分，也可以寫作結構來分析文章內容，也就是依需求將內容重點分解出來，而莊德明就利用心經科文識為心經的知識結構，這些結構的內容也是由古代學者詳讀心經後所歸納出來的結果。

而在論文研究的過程中，閱讀者可以依閱讀後所歸納後的重要內依樹狀結構進行編輯，因為每個人閱讀後所理解的方向不一定相同，所以使用者可進行個人知識樹的編修，再決定是否分享給予其他使用者參考，圖 4-10 所示為學習者在閱讀網頁注釋相關文獻後所得到的內容概要進行知識樹內容編修所得到的結果。

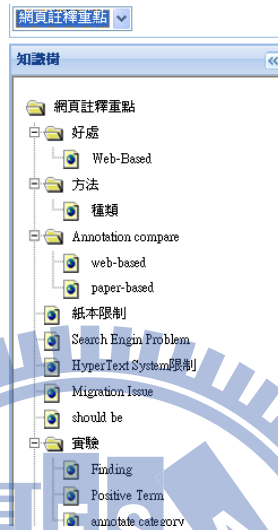


圖 4-10.知識樹內容

4.2.3. 文件對映功能

1. 新增文件對映

使用者先選擇知識樹節點後，再選取文件段落後，執行對映功能中的產生對映後，系統即讓所選取的節點與文件段落產生關聯，如圖 4-11；一個知識節點可以對映至文件中不同的段落，這部分解決了中文文獻系統中一個知識節點只能對映同一文件中唯一段落的限制；而知識樹節點可以與不同文件中的不同段落產生對映關係，讓知識樹與文件之間的關係產生不同的關聯性。

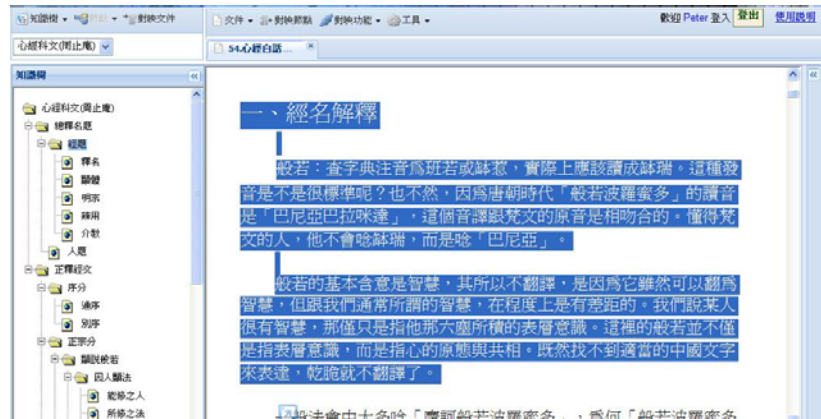


圖 4-11.新增文件對映

2. 節點對映文件

使用者在選擇知識樹節點後，執行對映文件後，系統右方的對映名單區塊會顯示在本文件中有針對所選擇的節點有進行對映的名單，選擇人員後系統就會將該使用者所對映的段落文字以灰底顯示，如圖 4-12；使用者就可以由此功能來比較不同使用者對同一份文件對同一知識節點對映的異同。

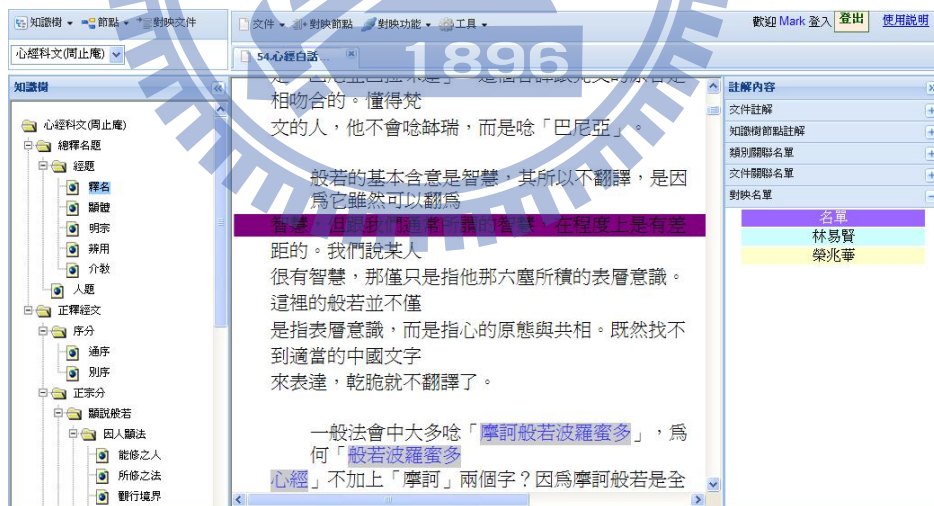


圖 4-12.節點對映文件

3. 文件對映節點

使用者選取文件中任何段落後，執行「對映節點」功能後，系統會將此文件有被使用者對映的節點以黃綠色底色顯示，如圖 4-13 所示，知識樹為詩品所定

義的種類，而文件內容為許渾的唐詩哭楊攀處士，使用者選取哭楊攀處士後，所對映到的節點為縝密與悲慨類型的詩品；文件中一段落可以對映至多個節點，此功能也解決了中文文獻系統中，同一文件段落只能對映至唯一節點的限制。

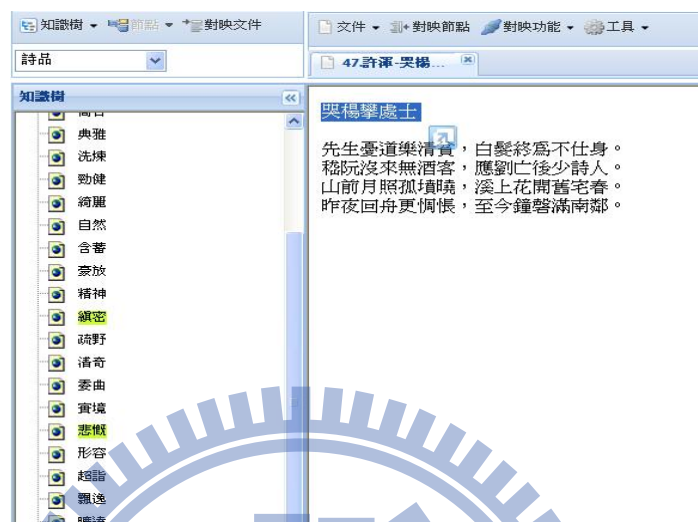


圖 4-13. 文件對映節點

4. 對映的交互運用

使用者可以透過對映的功能，遊走於不同文件與不同知識樹間，再加入使用者之間的連結，讓使用者、文件與知識樹產生多元件網路關係，也實現了超文件跳躍閱讀的特性。

首先，使用者可以透過文件對映節點功能由 A 文件經由知識節點連結至 B 文件，兩份會對映至相同知識樹節點的文件內容代表存在著相同的性質，如圖 4-14 所示，使用者可選擇心經法月版本的內容，對映至心經知識樹中的節點為總釋名題，然後再由節點總釋名題去尋找到玄奘版本中所對映到的內容後使用者就可以依此去比較不同版本之內容的異同，如圖 4-15。



圖 4-14.心經法月版本對映

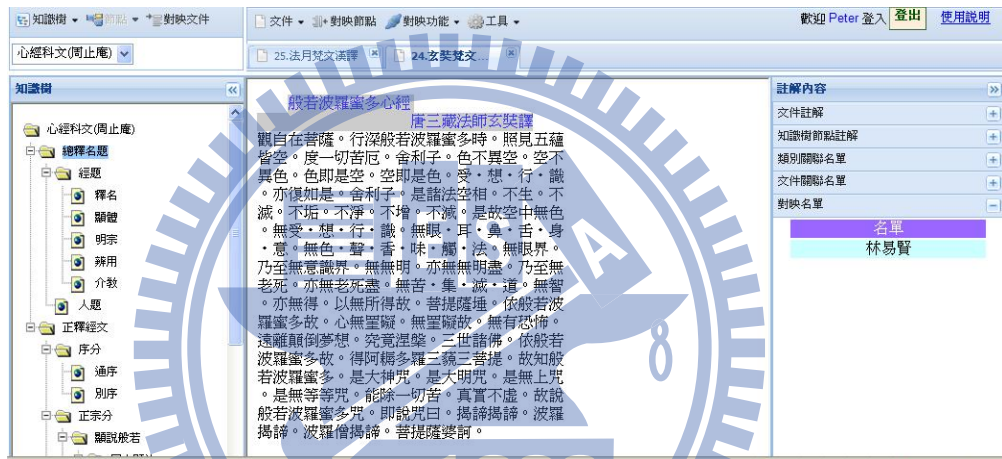


圖 4-15.心經玄奘版本內容

其二、使用者除了可以分析文件內容的異同，就可以比較文件結構的異同，如圖 4-16 所示，通序節點可以對映至法月版本的內容，但在玄奘版本無法找到對映，如圖 4-17，代表兩個不同版本的文件在周止庵科文的結構中，是有差異性的。

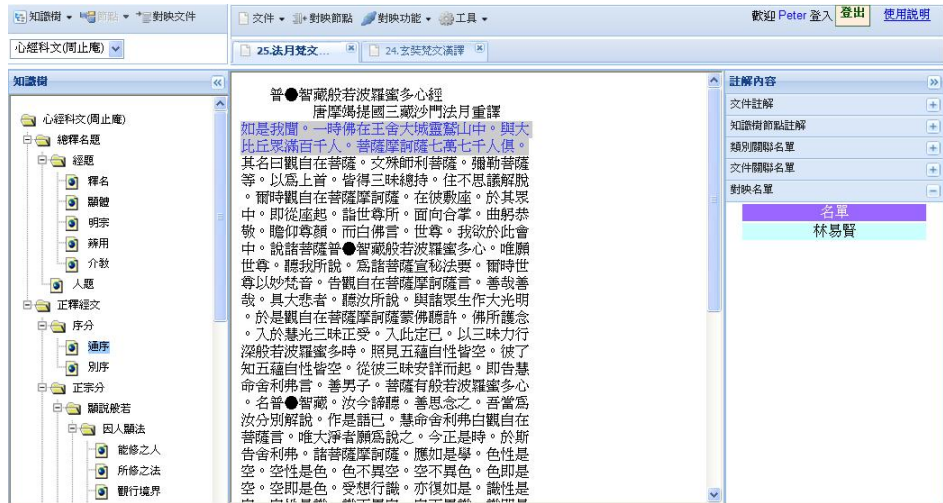


圖 4-16.通序節點對映心經法月版

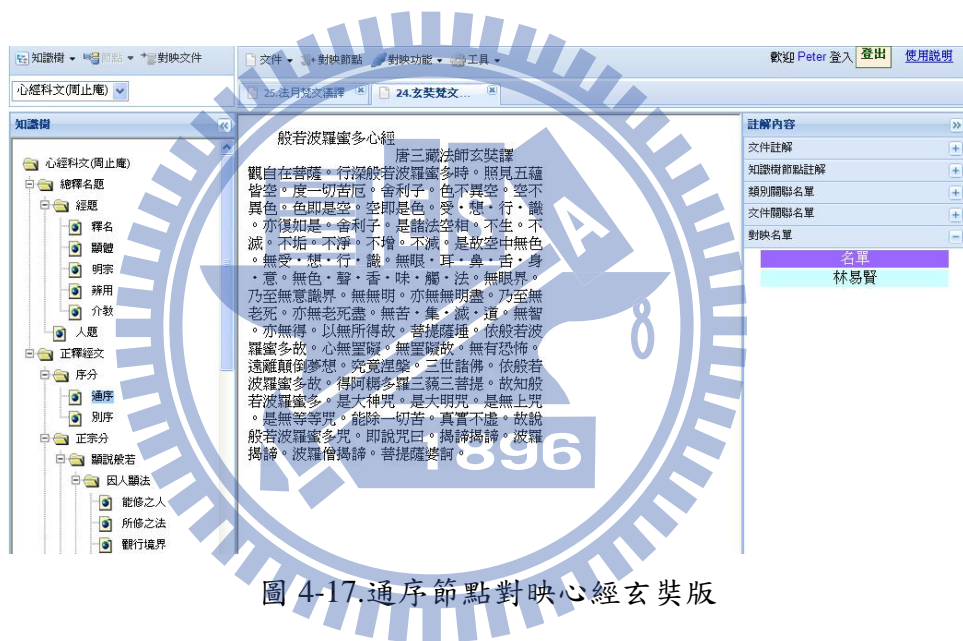


圖 4-17.通序節點對映心經玄奘版

另外在不同使用者詮釋的部分，每個人對文件內容的觀點不一定相同，除了可以針對內容建立不同的知識結構外，還可以針對特定的知識結構進行不一樣的對映，如圖 4-18 與圖 4-19，表示兩位使用者在同一份知識節點中所對映的同一份文件是有不同見解的，使用者 A 在心經玄奘版認為「照見五蘊皆空，度一切苦厄。」與知識節點「觀行境界」是相對映的，但使用者 B 所認為的「觀行境界」只包含了「照見五蘊皆空」，由此可見兩位使用者對玄奘版本的理解是有所差異的。



圖 4-18.使用者 A 的對映內容



圖 4-19.使用者 B 的對映內容

系統在詩品的應用方面，使用者可透過知識節點尋找文件內對映的內容，如圖 4-20 所示，使用者選擇了典雅節點，尋找到了唐詩中屬於此類的詩，就可以比較不同作者在同類詩品中的著作內涵；使用者也可透過典雅節點，找尋到司空圖二十四詩品本文內容，如圖 4-21 所示，了解司空圖對不同詩品所含有的意義為何。

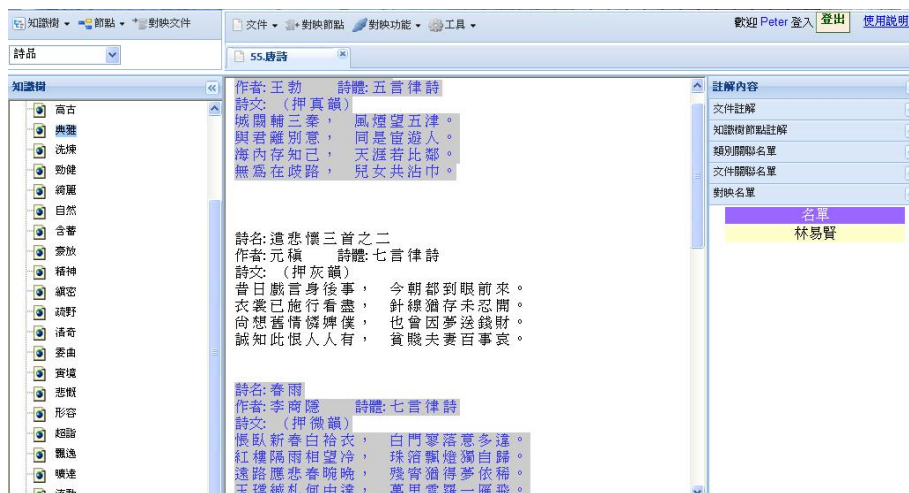


圖 4-20.典雅類唐詩對映

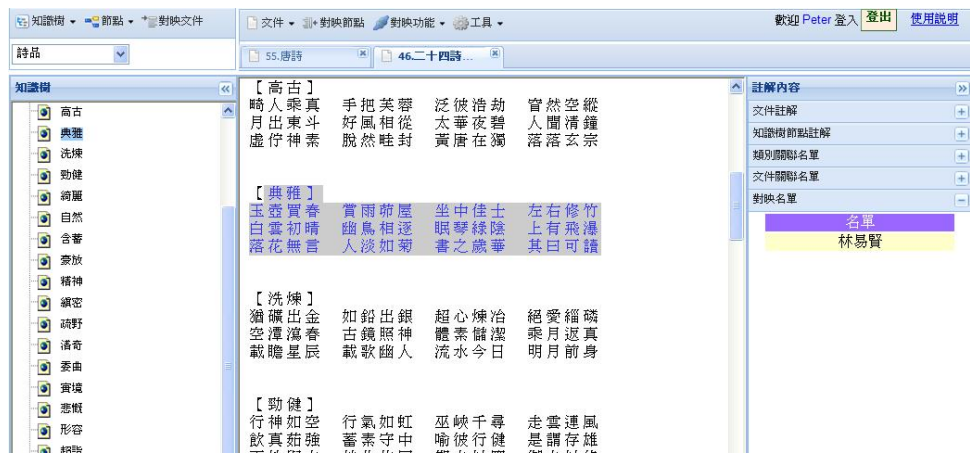


圖 4-21.對映詩品本文

透過文件對映功能的應用，使用者可經由閱讀後歸納後所建立的知識節點與文件段落的對映後產生的連結，在日後閱讀時能快速的找到所對映的內容，達到文獻[6]中所提到的資訊再尋找與協助讀者回憶內容的功能；而每一個知識節點也可以對映至不同文件段落，使用者可以將文件中某一段落透過知識節點連結至其他文章相關段落，加強超連結是連結一份文件，使用者需要重新閱讀後才能找到相關資訊的問題。

系統應用在論文研究方面，在研究的過程中通常會收集相關性質的文章進行閱讀，而這些文章多會存在著關聯性，使用者就可以透過系統將閱讀文章時所得到的概要建立成知識樹結構，再利用節點與不同文章中的相關段落進行對映，讓這些文章間產生關聯性，往後使用者在整理或重新閱讀文章時就可以由知識樹結構快速的尋找文章的相關內容，如圖 4-22 所示，使用者利用知識節點來載入相關文件，就可以過濾掉未被此節點所連結的文件；載入相關文件後可以再利用對映文件功能快速找尋到相關段落直接閱讀比較，如圖 4-23 與 4-24 所示，再透過注釋功能了解當時閱讀時的想法。

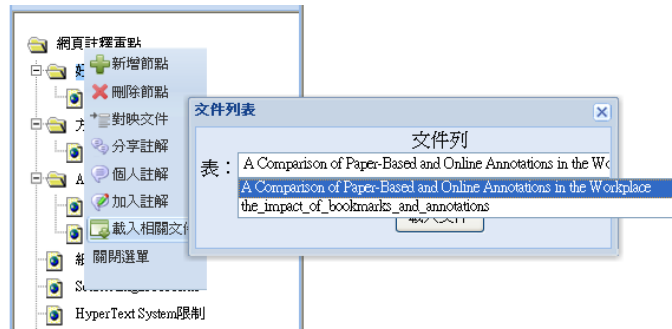


圖 4-22. 節點載入文件

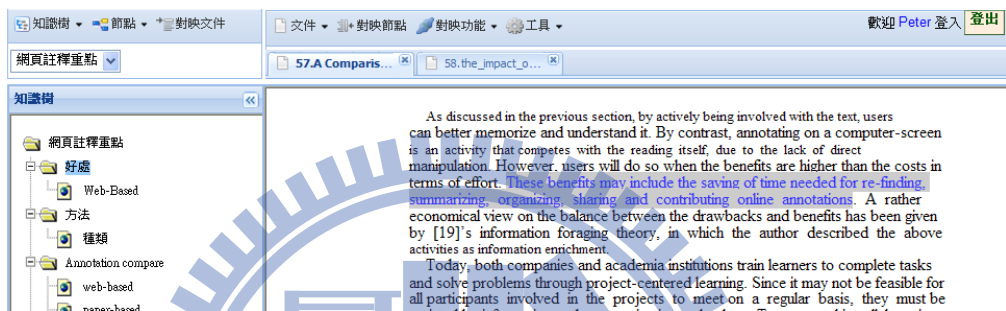


圖 4-23. 節點對映文件段落(1)



圖 4-24. 節點對映文件段落(2)

4.2.4. 文件注釋功能

1. 新增注釋

使用者可以針對知識樹節點進行注釋，系統提供了內容編輯器給使用者進行注釋編寫，編輯器可以建立超連結與外部網頁或文件的連結；使用者可以個人需求將注釋內容分享給予其他使用者或只供自己使用；如圖 4-25 所示。



圖 4-25.加入節點註釋

在文件內容的部份，使用者可以選取文件任一段落進行內容註釋，利用內容編輯器來分享或建立個人註釋，系統會將使用者所選取的內容視為關鍵文字。如圖 4-26 所示。



圖 4-26.文件內容註釋

2. 顯示註釋

使用者可選取知識樹節點後在右方知識樹節點註解區塊選擇要閱讀的為分享的註解或未分享的个人註解，如圖 4-27 所示。

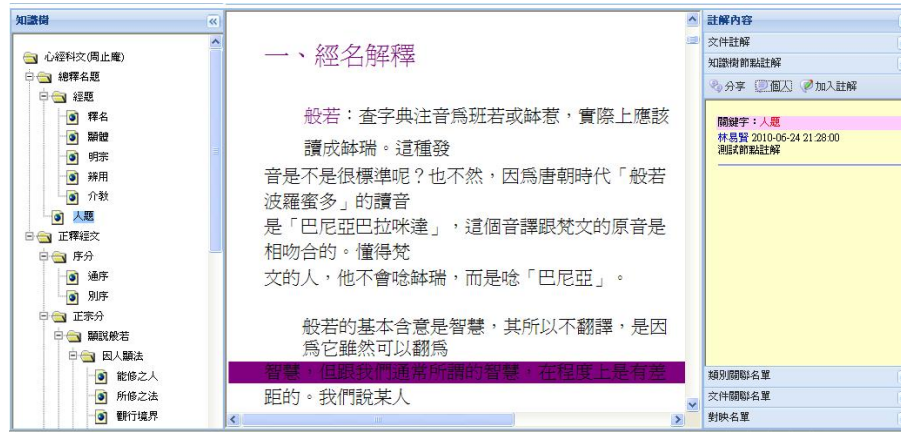


圖 4-27.顯示節點註解

使用者在選取文件後，如果屬於一般使用者，只能查詢分享註解；登入使用者則可查詢個人註解或分享註解，使用者在點選分享或個人註解後，系統則會將文件內容中被注釋過的文件以不同顏色表示，使用者選取文件中被注釋文字後，在註解內容區塊即會顯示注釋內容，因被文件中被注釋的文字可能產生重疊現象，所以在註解區塊以不同關鍵字作文區隔，如圖 4-28 所示。

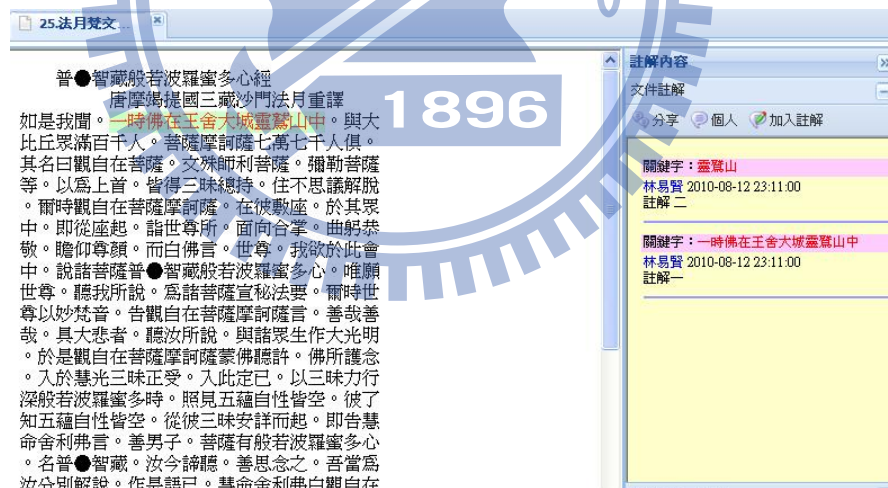


圖 4-28.顯示文件內容注釋

4.2.5. 個人收藏管理功能

使用者在登入後可以管理個人收藏，目前系統在此功能提供文件與知識樹收藏管理，使用者選擇類別後，系統會顯示在此類別下個人所收藏的文件與知識樹，如圖 4-29 所示。

使用者在上傳文件後預設為非公開文件，使用者可以透過此功能公開上傳文件或刪除個人未公開文件；使用者也可以此介面刪除個人收藏，但如果是個人已公開之文件，可能已有許多使用者將資訊建置於此文件上，避免資訊遺失，系統不提供刪除功能。知識樹的管理方法與文件相同。

此外，使用者在選擇類別後可以在工具選項中載入我的收藏，如圖 4-30 所示；系統只會將個人收藏的文件與知識樹載入列表中，過濾掉其他文件，避免過多的資訊產生。

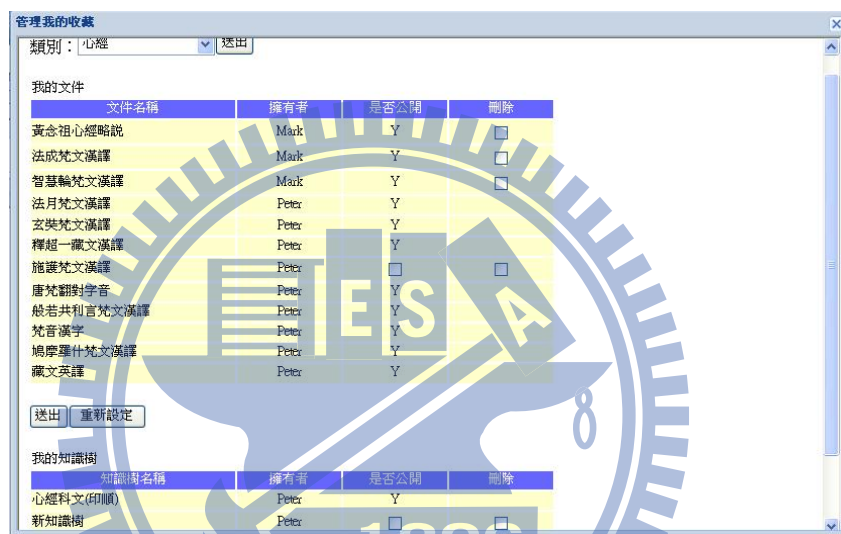


圖 4-29.管理我的收藏

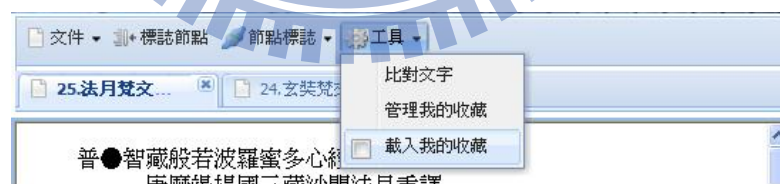


圖 4-30.載入個人收藏

4.2.6. 文件關聯圖

使用者在建立好文件與節點的連結後，系統會將使用者、文件與節點間的關聯，透過文件關聯圖來顯示，讓系統物件之間的關聯能透過此功能讓使用者一目瞭然。如圖 4-31 所示。

第五章、 結論與未來研究

本章總結整篇研究論文並提出未來可能的研究方向。5.1 節主要提出本研究的結論及在研究學習上的貢獻。5.2 節提出在論文研究的過程中因受限於本身或環境限制而無法進一步討論的相關議題，由系統的角度，提出不足之處或是可以精進及擴充之處做為未來相關之研究。

5.1. 研究結論與貢獻

本研究建置了一個平台讓使用者可以在文件上加上注釋而不變更文件內容，系統並讓文件之間透過知識結構產生關聯性，讓使用者可以在不同文件中遊走；因為注釋功能可以留下使用者所閱讀後所下的注釋，這對於後來的閱讀者可以承襲先前閱讀者的內容，更快的了解文件內容，經過不同使用者的加注，讓原來的文件不在只有本文，而讓關於文件的相關訊息可以不斷的增長，讓使用者可以在閱讀文件時可以得到許多可用的資訊，在學習上可以省去許多尋找資料的時程；閱讀者也可以針對文章內容進行個別的詮釋，並利用詮釋資料與其他閱讀者所詮釋的內容進行比較，也可以透過內容標誌連結至不同文件的相關段落，進行不同文件之間的比對。

總結來說，本研究主要的貢獻有：

1. 完成建置文件注釋系統，並將內容標誌的概念設計於系統中，方便讓使用者可以留下對文章的不同詮釋與見解，並透過網際網路分享予其他使用者。
2. 系統中利用內容標誌來連結不同的文章，讓使用者透過連結，尋找到更多相關的文件，而連結的內容是以段落為標的，讓使用者直接閱讀關聯性較強的段落

5.2. 未來研究

總結目前的研究心得與結果，發現還有許多議題可以進一步研究與修正，對於本研究的未來發展可朝下列方向進行：

1. 聯結圖書館資料庫

在文件方面，可以與圖書館資料庫進行連結，讓使用者可以透過圖書館更容易取得資料。

2. 加入數位學習的功能

本研究的目的是協助使用者除了資訊尋找的便利性外，也能讓使用者學習古文新知能更快速且更能上手，雖然目前已有許多文獻肯定了注釋工具對學習上的助益，但因時程的關係未能將數位學習相關功能建置完成，未來數位學習環境的建置更能讓顯現系統對使用者的幫助。

3. 多元的檔案內容

本研究目前只針對網頁文件進能注釋，而目前市面上已有不少工具可以針對 PDF 檔案進行注釋動作，如果未來能將 PDF 或其他相關文件類型納入系統中，勢必能將系統功能揮發更大的效益。

4. 加強社群功能

本研究目前只依文件關係將使用者進行分類，未來可以將系統連結至 Facebook 或 Plurk，讓使用者間的互動可以更密切。

5. 內容過濾與分類

因為系統是屬於開放平台，每個使用者均能上網提供內容注釋與內容，系統因為沒有進行良好的內容過濾機制，將會導致未來內容顯示的雜亂，

如：文件內容被所有注釋標示給佔據，而帶給使用者太多的雜訊，也是未來可以改善的地方



參考文獻

- [1] 謝清俊 (民國 96 年). "後設資料與內容標誌(上)." 佛教圖書館館刊 第 45 期: 頁 51-59。
- [2] 同[1], 頁 52。
- [3] 王梅玲(1995年7月), 「內容賞析資料庫初探」, 中國古籍整理研究出版現代化國際會議。
- [4] Nelson, Theodor H: Transliteration: "A Humanist Format for Re-Usable Documents and Media",2007. <http://transliteration.org/>
- [5] 陳昭珍(1994年), 「古籍超文件全文資料庫模式之探討」, 國立台灣大學圖書館學研究所, 博士論文。
- [6] 同[3], 頁 1。
- [7] 同[5], 頁 1。
- [8] Rune Dalgaard (2001.8), "Hypertext and the scholarly archive: intertexts, paratexts and metatexts at work", Proceedings of the twelfth ACM conference on Hypertext and Hypermedia.
- [9] 同[5], 頁 5。
- [10] Nelson, Theodor H: Transliteration: "A Humanist Format for Re-Usable Documents and Media",2007. <http://transliteration.org/>
- [11] Nelson, Theodor von Holm. "Xanalogical Structure, Needed Now More than Ever: Parallel Documents, Deep Links to Content, Deep Versioning, and Deep Re-Use." ACM Computing Surveys 31(4), December 1999
- [12] 同[1], 頁 52。
- [13] 同[1], 頁 55-56。
- [14] 同[3], 頁 1。
- [15] 同[1], 頁 53。
- [16] 同[1], 頁 57-58。
- [17] 謝清俊、莊德明, 古籍校讀工具「中文文獻處理系統」的設計, 中國古籍整理研究出版現代化國際會議·北京 1995年7月22-24, http://cdp.sinica.edu.tw/paper/1995/19950722_1.htm.
- [18] 莊德明(1995年9月), 「以《心經》為例說明如何利用計算機處理佛經的多版本」,

佛教圖書館館訊,第三期。

- [19] 同[5], 頁 4。
- [20] Chong, N. S. and Sakauchi, M.(2001). “Creating and sharing web notes via a standard browser. “,In Proceedings of the 2001 ACM Symposium on Applied Computing (LasVegas, Nevada, United States). SAC '01. ACM Press, New York, NY, 99-104.
- [21] P.-L.P. Rau, S.-H. Chen and Y.-T. Chin (2004),”Developing web annotation tools for learners and instructors”,Interacting with Computers, 16(2), pp. 163_181.
- [22] 同[20], 頁 2。
- [23] Kawase, R., Herder, E., Nejd, W. (2009), “A Comparison of Paper-Based and Online Annotations in the Workplace.”, In Proceedings of the European Conference on Technology-Enhanced Learning (EC-TEL 09)
- [24] Hoff, C., Wehling, U., & Rothkugel, S. (2008). “From paper-and-pen annotations to artefact-based mobilelearning”. Journal of Computer Assisted Learning.
- [25] 同[23], 頁 13。
- [26] I. Glover, Z. Xu, and G. Hardaker (2007), “Online Annotation – Research and Practices”, Computers & Education, vol. 49, no. 4, pp. 1308-1320.
- [27] Ricardo Kawase, G. P., Eelco Herder, Wolfgang Nejd (2010). “The Impact of Bookmarks and Annotations on Refinding Information.”, Conference on Hypertext and Hypermedia Proceedings of the 21st ACM conference on Hypertext and hypermedia: 29-34.
- [28] 同[21], 頁 5。
- [29] Henrik Eriksson(2007),”An Annotation Tool for Semantic Documents”,In Proc. of the 4th European Semantic Web Conference, ESWC 2007, Innsbruck, Austria
- [30] 計惠卿, 吳. (民國97年). "從網路空間屬性談網路學習社群的特質." 臺北市終身學習網通訊 41期: 頁29-34。
- [31] Awodele, O., Idowu, S., Anjorin, O., Adedire, O., & Apkore, V. (2009). “University Enhancement System Using a Social Networking Approach: Extending E-learning.”, Issues in Informing Science and Information Technology (IISIT). Volume 6, pages 269-283.
- [32] 同[30], 頁 5。
- [33] Jones, P. M. (2001). “Collaborative knowledge management, social networks, and

organizational learning.”, In M. J. Smith & G. Salvendy (Eds.), Proceedings of the 9th International Conference on Human–Computer Interaction (pp. 306–309). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

[34] 同[33]。

[35] 同[30]，頁 3。

[36] <http://www.javascript.com/> , Javascript, available at 2010.08.27.

[37] 陳會安、Javascript網頁製作徹底研究-第二版，旗標，台北市，民國92年。

[38] <http://www.sencha.com/> , ExtJS , available at 2010.08.27.

[39] <http://www.python.org/> , Python, available at 2010.08.27.

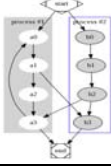

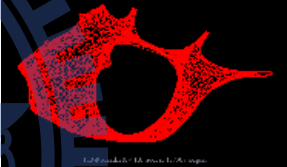
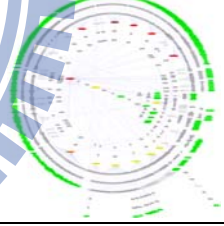
[40] 蕭世文、精通Python，文魁，台北市，民國92年。

[41] 黃書逸、Python 3技術手冊，基峰，台北市，民國98年。

[42] <http://www.graphviz.org/> , Graphviz , available at 2010.08.27.



附錄一 Graphviz圖形類型表

類型	說明	範例
Dot	階級式結構為主，圖形內容連結是有相同方向性。	
Neato	試圖將能量函數(Energy function)最小化，等同統計中的多維度排列。	
Fdp	與 Neato 類似，實作了 Fruchterman-Reingold heuristic。	
Sfdp	加強 Fdp 中較大的圖形類型。	
Twopi	雷達式的佈局方式，適合同一中心的多層環狀結構。	
circo	環形佈局，適合多個環狀結構與電信式網路。	