

第三章 SCORM 平台上之網路書籍模式設計

本章將說明以 SCORM 平台為基礎的網路書籍模式之基本概念與詳細架構。第一節討論網路書籍模式的定義及詳細的架構說明；第二節則介紹引入 SCORM 規範與技術後必須進行之對應與修正，使網路書籍模式可以符合 SCORM 課程整合模型規範，並說明在 SCORM 平台的規範之下建置符合 BOD 模式理論之網路書籍的步驟；第三節針對網路書籍的索引頁與外部連結頁提出本研究發展出來的自動頁面產生機制；第四節則整理出 SCORM 平台上網路書籍模式的基本架構模式與內含的各類頁面之型態與定義。

第一節 網路書籍模式概念

本節主要介紹網路書籍模式的動機與相關理論背景，首先提出邏輯網頁的概念與網路書籍模式理論，然後定義系統的使用角色並描述各種使用者的使用情境和任務，最後提出網路書籍的組成結構及各類型頁面型態之說明。

3.1.1 網路書籍模式與邏輯網頁之理論與定義

網路的問世與普及不但改變人類的許多生活模式，更進一步對傳統以紙本書籍作為知識傳承與文化遞嬗唯一媒介的地位產生衝擊，書本不再侷限於

紙本的平面呈現方式，取而代之的是在網路上流傳的各種電子檔案格式，各類資訊透過網路這個便捷的管道零時差、跨地域地傳播到世界各地，而在多媒體技術發達的影響之下，呈現在使用者面前的資訊有了更生動活潑的面貌，聲音、圖像、文字可以同時存在於「書本」之中，不但有助於提升學習者的接收意願，同時也可以增進學習效率。

在這樣的前提之下，我們認為新型態的書籍概念將包含以下特點：內容來自全球資訊網這個龐大的知識體、即時且互動式的瀏覽行為、結合多媒體的呈現方式並可以動態組成滿足瀏覽者個別學習需求的教材。我們將此種書本的形式稱為「隨取書籍」(Book-On-Demand, BOD)【9】，而依照 BOD 模式所建置的書籍則稱為網路書籍(Web Book)【3】【8】。在此模式之下，教材的編寫者角色不只是再侷限於教師，只要透過適當的工具，每位瀏覽者都可以依照自己的需求來建置書籍，而其內容的來源為龐大且豐富的網際網路知識庫，而在網路自由與分享的基本理念上，更可允許使用者將自己編輯好的書籍放到網路上與來自世界各地的其他使用者分享。BOD 模式的基本概念模式如圖 3-1 所示。

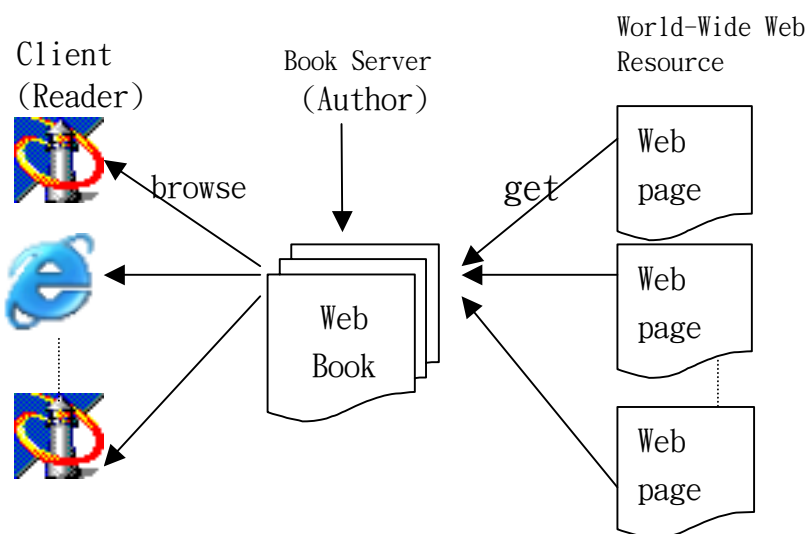


圖 3-1 Web book 基本架構

BOD 模式分為五個主要部分，分別是：資源檔案、書籍伺服器、作者、讀者以及網站管理員。其中作者編輯個別化網路書籍以分享給網路上的讀者，網站管理員則負責書籍伺服器的運作正常。資源檔案則位於全球資訊網中。以下針對上述五個部份分別討論：

1. 網站資源(Web Resource)：以全球資訊網各個網站的網頁檔案為主，提供網路書籍的作者編寫網路書內容的知識與資訊來源。
2. 書籍伺服器(Book Server)：書籍伺服器的功能類似數位圖書館的核心引擎，管理由作者蒐集資源檔案編輯而成的網路書籍，在讀者閱讀網路書籍時，提供網站資源快取下載，並提供讀者一般書籍的各種服務功能，如：內容架構、索引、字串搜尋。

3. 作者(Author)：負責編寫網路書籍，作者可利用書籍伺服器提供的書籍編輯器來整理各個網站的網頁檔案，並提供邏輯式的書籍結構。書籍內的資料來源可由作者提供或是蒐集資源檔案而得。
4. 讀者(Reader)：讀者在閱讀網路書籍時，以瀏覽器連接到書籍伺服器即能閱讀作者所建立的網路書籍。
5. 網站管理員(Web Master)：網站管理員負責的工作類似於數位圖書館的管理員，其管理書籍伺服器相關工具軟體的運作以及個別化網路書籍的上架和每本網路書籍各種快取參數的設定並負責書籍伺服器的正常運作。

個別化網路書籍模式的運作可以區分為兩個階段：第一階段為編輯階段(Authoring Phase)，由網路書籍作者透過書籍伺服器編輯網路書籍；第二階段為閱讀階段(Reading Phase)，由讀者透過書籍伺服器閱讀各種個別化網路書籍。網路書籍編輯階段的運作如圖 3-2 所示，作者透過書籍伺服器內的書籍編輯器(Book Editor)的操作，輸入網際網路內各網站的網址，書籍編輯器即下載網址指示的網站入口以下的實際網頁結構和檔案。這些檔案以實體網頁結構(Physical Web Page Structure)將各個網頁檔案整合成實體網頁(Physical Web Page)。作者透過書籍編輯器的協助，建立個別化網路書籍的邏輯網頁結構(Logical Web Page Structure)以及產生邏輯網頁(Logical Web Page)的檔

案。在編輯好相關檔案之後透過書籍伺服器上傳到書籍目錄(Book Archives)內完成個別化網路書籍發行上線的作業。

作者在編輯網路書籍時，就像在撰寫傳統書籍一樣，也要提供書本內容的章節結構和索引。其章節結構是一種階層式的知識組織，可以幫助讀者掌握書本內的知識結構，而索引則可以幫助讀者快速地找到專有名詞或關鍵字在書本中的位置。

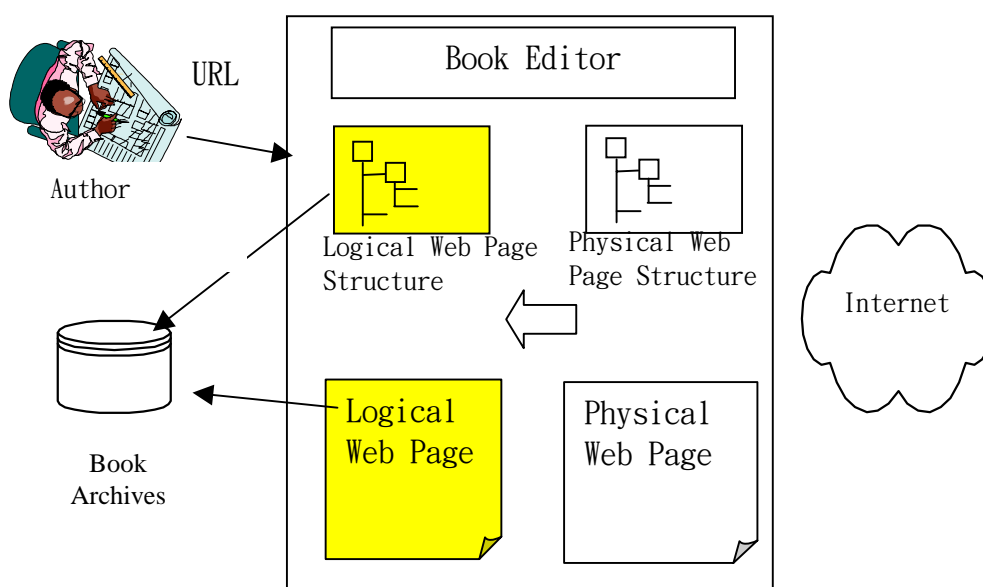


圖 3-2 個別化網路書籍編輯作業

在網路書籍閱讀階段，讀者則是透過書籍伺服器來閱讀由作者編輯完成發行上網的個別化網路書籍。讀者是以作者提供的邏輯觀點來閱讀網路書籍的內容，因此讀者不再需要使用網址來閱讀網頁，而是使用邏輯網頁頁碼(Logical Page Number)到書籍目錄去找到對應的位址。讀者在閱讀網路書籍時，可以按照作者編輯的章節結構的呈現來閱讀，且由於網路書籍也同樣提供傳統書籍所具有的書本章節內容架構和索引，使其可兼具有傳統書籍架構

嚴謹和超文件網頁瀏覽方便的優點，並可避免產生一般使用者在使用全球資訊網時產生的缺失。

3.1.2 網路書籍架構

讀者在書籍伺服器中挑選出來閱讀的網路書籍，其呈現在讀者面前的架構其實與該書籍之來源網站原有的架構不盡相同，讀者所看到的只是經過作者重新編排架構後的一個或多個網頁所構成之網路資訊集合。而為了保留傳統書籍閱讀的優點以及情境，Web Book 採用單一入口的方式，讀者一旦選定所要閱讀的章節並進入該章節開始進行閱讀活動後，只能進行線性的瀏覽動作，亦即只能往前或往後一個頁面，且書籍本身採用單一入口的限制，讀者只能從書籍的首頁進入閱讀。除了外部連結可讓讀者連至書本以外位於全球資訊網上的網頁外，書籍內部的連結呈現封閉狀態。在比照傳統紙本書籍的組成架構並且不失網際網路瀏覽優點的考量下，本研究將 Web Book 的構成網頁型態分為八種，分別是書籍目錄網頁、網路書首頁、內容架構、索引、外部連結網頁、網路書狀態網頁、字串查詢結果網頁以及網路書內容網頁。圖 3-3 為網路書籍內各種網頁連結的狀態，讀者以書籍目錄首頁為主要進入點以選擇欲瀏覽的網路書，並進入書籍開始閱讀。

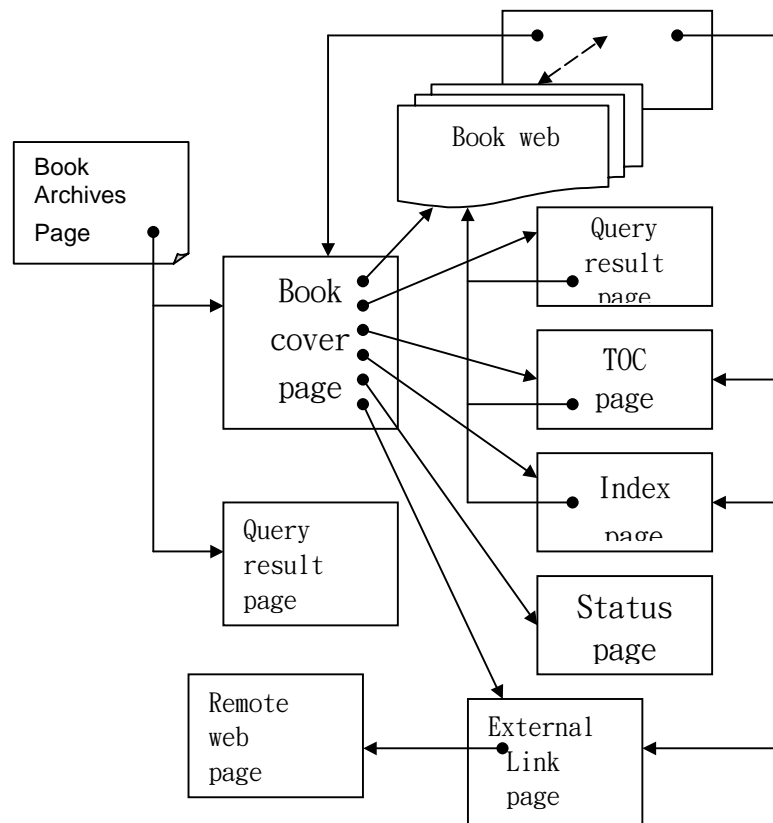


圖 3-3 網路書籍的網頁結構和連結狀態

以下我們將簡單說明上述的八種網頁型態的內容及其功能：

1. 書籍目錄網頁：讀者進入書籍伺服器的第一個網頁，記錄伺服器中所有網路書籍首頁的連結，並提供讀者查找書籍的功能。
2. 網路書首頁：此為網路書籍的首頁，內含網路書中的章節與網路書狀態網頁連結，並提供網路書的字串查詢功能。
3. 內容架構網頁：以階層式架構來顯示網路書的內容架構，並可點選以連結網路書中任一網頁。
4. 索引網頁：由匯集作者所編輯的索引編成，以關鍵字串的 ASCII 碼遞增排列，並顯示具有該關鍵字的網頁主題及頁數。

5. 外部連結網頁：將網路書中所有的外部連結集合起來，以頁數的順序排列列出每頁的外部連結。讀者點選時則以新的瀏覽器視窗開啟連結。
6. 網路書狀態網頁：描述網路書的狀態，包括作者資料、網路書建立日期、網路書更新狀態等。
7. 字串查詢結果網頁：查詢範圍可分單一以及所有網路書。在網路書首頁內查詢該網路書的網頁內容時，以頁數遞增順序列出符合查詢字串的網頁標題；而在書籍目錄中查詢所有網路書的網頁內容，則以網路書名稱為主鍵、頁數為副鍵，分群列出符合查詢字串的網路書名稱及網頁標題。
8. 網路書內容網頁：為各網路書籍內容的主體結構，保留指向網路書內的連結，而將外部連結抽出。

第二節 網路書籍教材的 SCORM 轉換規劃

這一節中將說明 SCORM 平台上的個別化網路書籍模式，以及在網路書籍模式中引入 SCORM 標準所需顧及的考量。首先針對個別化網路書籍模式運作時的兩個階段探討：作者在編寫網路書籍時的操作行為和先前所提的模式相仿，其教材的主要來源可以是取自網路上的資源，也可以是作者自己提

供的網頁內容或取自教材元件庫中的教材元件或自編的教材元件。在書籍編寫階段中，作者的任務是到元件庫中搜尋相關的元件，選取所需的內容編輯成網路書籍，元件的來源也可以從現有的書籍中取得，如果無法找到所需的教材元件，則可以由網路上下載取得所需的內容，自行編製成適合的教學元件。選定好書中所要包含的教學內容後，下一步是編輯網路書籍邏輯架構。在 SCORM 的規範中，課程物件的架構是透過清單檔(Manifest)加以描述，而其原本的概念與作法就是讓實體資源和邏輯架構的描述各自獨立，作者只需要將編輯好的清單檔和實體資源檔案一起打包成課程包裹後上傳至伺服器中，然後由伺服器管理員將書本匯入並向系統註冊，便可以完成書本上線的動作。SCORM 規範為了兼顧網路操作以及跨平台的考量，課程包裹的資源檔案本身必須是透過瀏覽器可以開啟的檔案格式，讀者只需要透過一般的瀏覽器連接到伺服器上選取所需要的課程書籍或課程並開始整個閱讀的學習活動即可，不再需要藉助特定的軟體或工具便可進行閱讀的動作。

而編寫 SCORM 平台上之網路書籍的步驟，本研究將其分為書頁下載、課程元件編製、課件打包以及書本上線四個進程，以下針對四個部分個別說明：

A. 書頁下載

陳連堦【3】提供的網頁下載策略中，雖然會將下載的網頁之間的連結關係重新定義，但仍保留某種程度的相互關聯性。然而在 SCORM 規範中，一旦 Run Time Environment 啟動一個 SCO 後，在該 SCO 的狀態為完成(Finish)之前，不可以再開啟另一個 SCO，SCO 本身也不允許執行啟動其他 SCO 的動作。所以為了避免下載的網頁在被製成課程元件後出現因為連結動作的執行而在該元件本身未完成被讀取狀態前就去啟動其他 SCO 的可能，本論文將網頁下載策略修正為只針對個別網頁的下載為主。下載程式對作者所提供的網頁位址所在的頁面進行剖析的動作，將與該頁面顯示有關的實體資源連結(如圖片、聲音或影像等)與檔案完全下載，完成後將其餘的連結全部取消，並將網頁中不必要的資訊(例如原網頁中的<script>、<style>等標籤)移除，然後將下載後的檔案間連結關係由絕對路徑參照改為相對路徑，再將檔案儲存至作者的電腦中，便完成了書頁下載的動作，暫存於作者電腦中的書頁必須等到編製成符合 SCORM 規範的元件後方可上傳到元件伺服器中。

B. 課程元件編製

SCORM 規範中，教材元件的實作方法是在現有的教材檔案上附加上一份 XML 描述文件，將檔案描述為課程元件。對於書籍中所需的教材內容，我們先進行分類的動作，再為各自的類別定義對應的物件包裝層級。說明如下：

1. Assets: Asset 是編輯教材所需的素材，其格式可由文字、圖形、聲音、網頁或任何可由瀏覽器讀取的電子檔案格式(如 Flash 檔)所組成。我們將網頁中圖片及其他媒體格式如聲音、影像等定義為 Asset。另外網頁中可以重複使用的知識，例如表格、分項列表資料等在各類型網頁中被重複使用的可能與頻率也高於該網頁的內容整體，故此種類型資料也定義為 Asset 層級。上述資料類型在依照 SCORM Content Aggregation Model 規範編寫出對應的 XML 描述檔後，編寫書籍時只需將所需的材料以 Asset 的方式加入清單檔 `imsmanifest.xml` 中即可。圖 3-4 為取自 Manifest 檔中關於 Asset 資源實體的描述區段。

```
</resource>
- <resource identifier="MANIFEST01_RESOURCE14" type="webcontent"
  href="content/pics/04_banner.GIF" adlcp:scormtype="asset">
- <metadata>
  <schema>ADL SCORM</schema>
  <schemaversion>1.2</schemaversion>
  <adlcp:location>content/pics/04_banner.xml</adlcp:location>
</metadata>
<file href="content/pics/04_banner.GIF" />
</resource>
- <resource identifier="MANIFEST01_RESOURCE15" type="webcontent"
  href="content/pics/arrow00.GIF" adlcp:scormtype="asset">
- <metadata>
  <schema>ADL SCORM</schema>
  <schemaversion>1.2</schemaversion>
  <adlcp:location>content/pics/arrow00.xml</adlcp:location>
</metadata>
<file href="content/pics/arrow00.GIF" />
</resource>
- <resource identifier="MANIFEST01_RESOURCE16" type="webcontent">
```

圖 3-4 Manifest 檔中 Asset 的描述區段

2. SCO：依據 SCORM 規範，一個 SCO 被啟動(Launch)後，則必須在狀態為完成後才能離開，且執行中的 SCO 不被允許啟動其他的 SCO，這與網路書籍模式線性閱讀的理念相符。在個別化網路書籍模式中，學習者進入一個章節後，只能以線性的方式(意即只能向前或向後換頁)進行閱讀動作，直到閱讀完畢才可以離開該章節轉到其他單元。為了兼具傳統書籍以及網路瀏覽的優點，章節中的各網頁允許保留某種程度的連結，只要在章節內呈現封閉狀態即可。故本研究將 SCO 對應至書籍概念的章節形式；如果章節中的頁面較多或者彼此有連結上的需要存在時，則為該章節加入一個根頁面(Root)，所有網頁的連結最後都將回到根頁面而不會有啟動到其他 SCO 的可能發生，加入清單檔時則將根頁面定義為 SCO，其下的頁面為 Asset，在書籍

的清單檔中關於此類章節元件的邏輯架構描述區段必須與各頁面採用的封閉樹狀連結架構完全對應。而後便可將單一頁面或者封閉樹狀的章節 SCO 附加到清單檔中。圖 3-5 為取自 Manifest 檔中關於某個章節資源實體的 SCO 描述區段。

```
<adlcp:location>content/ch4_S02.xml</adlcp:location>
</metadata>
<file href="content/ch4_S02.HTM" />
<file href="content/pics/arrow00.GIF" />
<dependency identifierref="MANIFEST01_RESOURCE15" />
<dependency identifierref="MANIFEST01_RESOURCE17" />
</resource>
- <resource identifier="MANIFEST01_RESOURCE4" type="webcontent"
  href="content/ch4_S03.HTM" adlcp:scormtype="sco">
- <metadata>
  <schema>ADL SCORM</schema>
  <schemaversion>1.2</schemaversion>
  <adlcp:location>content/ch4_S03.xml</adlcp:location>
</metadata>
<file href="content/ch4_S03.HTM" />
<dependency identifierref="MANIFEST01_RESOURCE17" />
</resource>
- <resource identifier="MANIFEST01_RESOURCE5" type="webcontent"
  href="content/ch4_S04.HTM" adlcp:scormtype="sco">
</metadata>
```

圖 3-5 Manifest 檔中章節 SCO 的描述區段

3. 內容架構：在原来的網路書籍模式中，包含了一個內容架構頁，用來顯示書籍中的內容架構。而在 SCORM 平台中，只要將所需的教材元件打包成課件包裹後，在顯示時採用選擇式 (Choice) 便可在瀏覽畫面的子視窗中顯現課程中所有元件的組成架構，所以作者只需在打包課程的過程中將架構定義完成就可以，不需再額外含入書籍內容架構頁。

C. 課件打包

依據 SCORM 規範，打包的動作主要是透過清單檔(Manifest)來描述元件的邏輯架構，將所需用到的全部元件檔案實體連同 Manifest 檔壓縮為課程包裹(Content Package)，便能成為一個完整的教學課程。在 Manifest 中分別描述了課程的邏輯架構和實體檔案分佈，以及邏輯元件(Item)與實體資源間的關聯性，一方面可以告知執行平台課程包裹中所包含的資源與分配狀況，方便管理與重用。另一方面位於架構(包在<organizations>與</organizations>標籤之間)區段中的課程邏輯架構與實體資源(<resource>與</resources>標籤之間)區段彼此充分互相獨立，亦即邏輯架構與實體資源的分配架構可以完全不相同，此一特色有助實踐網路書籍模式理論中邏輯式網頁的理念。在預設已存在教材元件庫之前提下，書籍作者可以按照個人所要呈現的學習情境進行書籍編寫的動作，將書籍內容架構寫入 Manifest 檔裡，並且詳細交代邏輯元件與實際資源物件間的關連，再將清單檔與實體資源檔案壓縮成課件包裹，便能建造出符合網路書籍模式的教材提供讀者閱讀。

D. 書本上線

最後將書本的壓縮檔上傳至符合 SCORM Run Time Environment 規範的管理系統，然後由管理者將書本匯入到執行環境中，向系統註冊該書的相關資料便可完成書本的發行。至目前為止，對於作者來說，已經完成了編輯書本的任務；對讀者來說，僅需以瀏覽器連結並登入書籍系統即可選擇任何一本網路書籍開始閱讀；系統管理者的任務則是負責將作者上傳的書籍註冊至系統中以供讀者選則，並維護線上書籍系統以及元件庫的正常運作，還要不定時將書中的元件更新到教材元件庫中，以幫助其他的書籍作者日後編寫書籍時可以有更多的素材可以使用，落實教材元件共享及重用的理想。

第三節 書籍自動索引機制的規劃

前文討論的網路書籍模式中，作者的編輯任務包含索引頁之編輯動作，然而由作者來編寫並將內容連結至書籍中的動作不但費時費事，也可能因為不當的連結動作導致違反 SCORM 平台規範的發生，本研究特針對此問題提出一套自動索引的機制來避免可能發生的問題並簡化作者編輯索引頁的工作。主要的概念如下：

在下載的網頁中，網路書籍作者可在文件中找出其認為比較重要的關鍵字，並將關鍵字的內容以標籤(Tag)的形式存於文件之中，在書籍編輯階段時，書籍編輯器會在一個 HTML 網頁被加入的同時剖析該份文件，找出作者所額外加入的標籤，然後將內容所標示的關鍵字句蒐集起來，自動產生索引頁，並在各個關鍵字後方整理出相關頁面的連結，使用者可以點選感興趣的關鍵字連結到該網頁中進行閱讀活動，進入該頁後則依照 SCORM 的規定直到完成該頁所在之 SCO 後才能再選擇其他單元繼續閱讀的活動。針對上述概念，我們定義出標籤<term tag="術語分類">以及<term tag="連結" description="連結描述">兩類。術語分類為一般傳統書籍中比較常用來列入索引的幾種類別，如人名、專有名詞等。

作者在使用網頁下載工具時，可選定所需加入之索引詞句，而後透過下載工具本身提供的功能將上述標籤加進下載的網頁中，由於目前瀏覽器對於無法解讀的標籤並不會進行特殊的顯示處理，所以加入這樣的標籤對於讀者在閱讀書籍時並不會造成任何影響。圖 3-6 為作者在網頁中加入<term>標籤的動作示意圖。

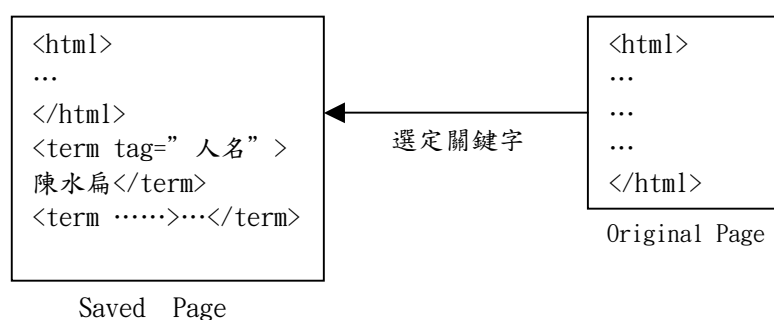


圖 3-6 加入索引標籤示意圖

在書籍編輯器必須提供的相關功能方面，則是在作者編輯書籍架構並且將實體資源加入清單檔時，書籍編輯工具對每次加入的頁面進行剖析的動作，先讀取<term>標籤的內容，如該標籤的屬性為 tag 則表示此標籤標示的字句是作者欲加入索引頁的關鍵字，此時書籍編輯系統會去查詢一張名為 ToT(Table of Terms, ToT)的表格，如果表格中已經存在相同的欄位，則將此標籤的所在頁面附加到表格中該關鍵字所在欄位的連結區之後；如表格中沒有相同的關鍵字存在，則新增一個欄位到表格之中。ToT 的形式概念表示如表 3-1 所示。

Term_attr	Term_catalog	Term_content	Link_address
Keyword	Name	Str_of_names	Target page
	Phrase	Str_of_phrases	Target page
	Keywords	Str_of_keywords	Target page
Link	Description	NULL	External URL

表 3-1 ToT 架構表綱要

當作者完成書本的編輯並且將所有的相關資源與清單檔打包時，書籍編輯系統的工作便是將 ToT 的內容寫到索引頁之中。從表 3-1 中可以看出，<term>標籤在設計的時候也考量到外部連結的需要，所以根據 ToT 所產生出來的頁面共有兩個，一是網路書籍中的索引頁，另一個則為外部連結頁面。由於原先的外部連結頁面本身已經是由各頁面中所收集來的外部連結組合而成，故本研究將<term>標籤所指示的外部連結定位為作者所額外提供的

相關參考資源，ToT 的內容只能作為該頁面的部份來源而非全部。由上述的

理念，書籍的編輯流程也有對應的改變，詳細流程如圖 3-7 所示。

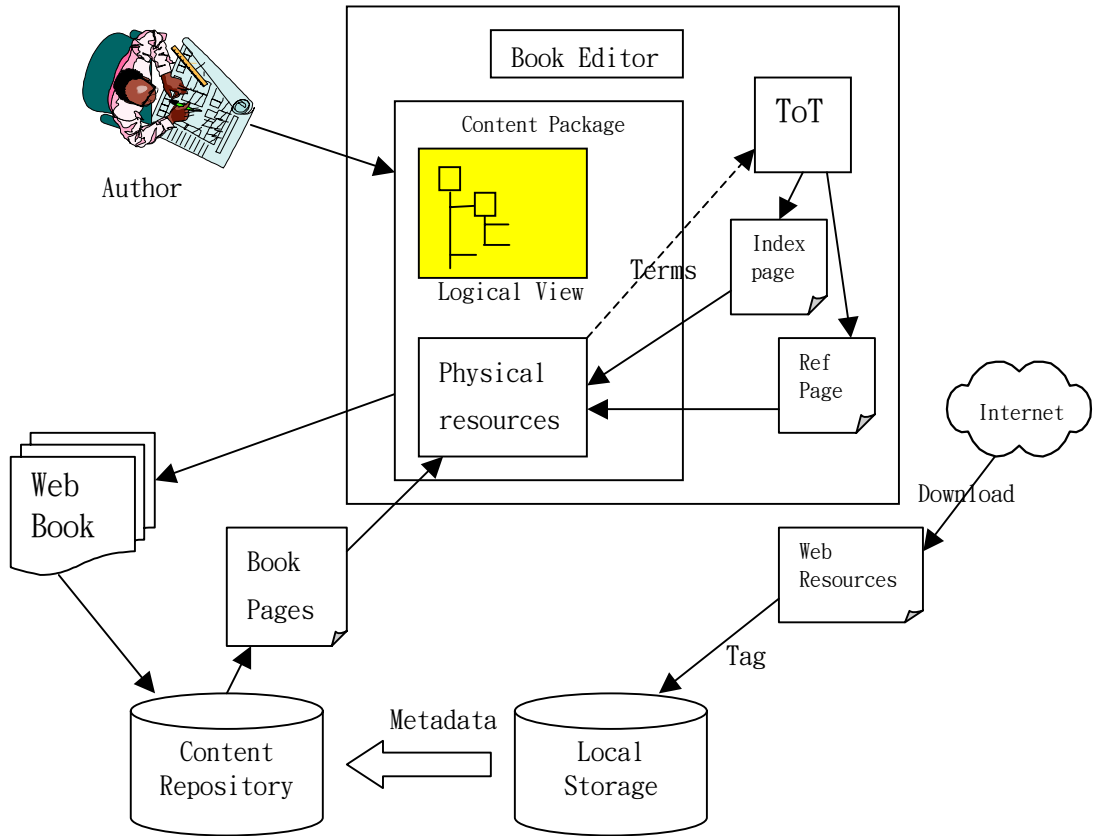


圖 3-7 SCORM 網路書籍編輯流程

第四節 SCORM 網路書籍模式架構

歸納前文的理念，本研究將網路書籍的內部網頁型態重新定義為書籍目錄網頁、網路書首頁、內容架構、網路書內容網頁、索引頁和外部參考連結頁等六種型態，各類網頁間架構模式如圖 3-8 所示。

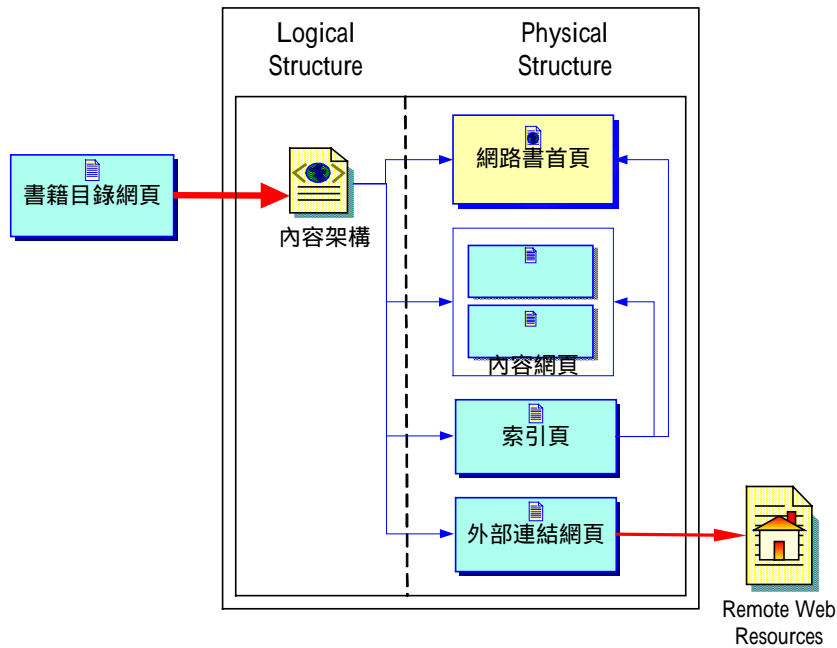


圖 3-8 SCORM 模式 Web Book 架構圖

各類型頁面的說明如下：

1. 書籍目錄網頁：此網頁為讀者登入書籍系統並進入教材選擇的項目後所呈現的網頁，網頁的內容為目前經由其他書籍作者編製完成並向系統註冊後的書本(Registered Course)，讀者可以選擇自己有興趣的單元後向系統確認，而後開始進入閱讀畫面。
2. 網路書首頁：為網路書籍的首頁，也是整本書中的第一個 SCO，該頁面在被啟動後會將使用者的資料存入系統資料庫中，例如帳號名稱以及啟動本課程的時間等，以方便日後對該讀者的學習歷程進行後續的分析或追蹤的動作。

3. 內容架構：在 SCORM 的教學平台規範上，將內容架構獨立在系統的子框架(Frame)中，並將該課程的架構以樹狀圖的型態顯示在框架內，所以網路書籍的作者編寫教材時填入清單檔中的邏輯架構可以由系統自行顯示出來，不再需要額外編寫書籍架構網頁。
4. 網路書內容網頁：僅保留指向網路書內的連結，而將外部連結抽出。如單一網頁為一完整之教學單元時，則將該網頁定義為獨立 SCO，而如該學習單元涉及數個網頁時，則將相關網頁編寫為封閉樹狀的形式，以根網頁為 SCO，下層網頁彼此之間的連結必須限於此封閉的樹狀結構中，不得連結至其他章節或學習單元。
5. 索引頁：由作者在個別頁面中加入<term>標籤，並填入所需要的詞句和該詞句的類別，放置於該頁面 HTML 文件之最後。作者編輯書籍的時候，可由系統自動抓取每個頁面的<term>標籤，在書籍編輯階段中存於 ToT 表格，當作者完成書籍編輯並確認儲存動作後，由書籍編輯器自動將 ToT 中的內容轉為索引頁，並將索引頁以 Asset 加入至網路書籍課程包裹中。頁面中的術語後有指向各相關頁面的連結點，以流水編號的方式呈現對應連結到各頁面，由連結點進入頁面後，讀者仍須待該頁面所在的封閉樹狀章節完成後再點選至其他章節進行閱讀動作，這樣的規劃可避免發生違規啟動 SCO 的操作行為，亦能符合網路書籍理論線性閱讀的基本理念。

6. 外部連結頁：同索引頁面的產生方式，其內容主要來自兩個來源，一為該頁面的原始來源位址，另一個則是作者加在各頁面中以作為補充之用的相關參考連結。讀者點選連結後，會將該連結開啟在新瀏覽器視窗中。