

數位內容格式轉換系統—從 SCORM 數位教材到 EPUB 數位出版品

張玄菩

邱翊

洪浩容

淡江大學資訊與圖書館學系

中央大學資訊工程學系

淡江大學資訊與圖書館學系

助理教授

研究生

研究生

musicbubu@gmail.com

joeyc0916@gmail.com

kota@hotmail.com.tw

摘要

近來電子書資源與應用藉由電子閱讀器的普及快速地席捲人們日常生活與閱讀習慣，也改變了傳統出版業的生態環境，數位閱讀嚴然已經成為一股不可抵擋的未來趨勢，因此從何產生優質的數位出版品內容即成為一個重要的議題。事實上在數位學習的發展與成果中，目前已經累積相當多的優秀教材與內容，但這些教材內容大多儲存在教材儲藏庫(Learning Object Repository)或學習平台(Learning Management System)中，由於在教材的設計與執行上必須搭配特定的學習環境與規範管理，因此在內容使用度上與能見度上都受限於平台的建置與策略的規範；另一方面，同樣是數位內容的電子書只要有電子閱讀器就能隨時隨地享受閱讀的學習與樂趣。從內容的權威性與嚴謹度來看，許多數位學習的內容教材往往也是教師集成個人教學與研究之心血，在未來個人出版的趨勢與機會下，以個人數位出版品呈現會是未來教師另一個具體成就的表現。

因此本研究提出一個數位學習教材與數位出版品的格式轉換系統，可將符合 SCORM 標準的數位學習內容教材快速轉換成符合 EPUB 標準的數位出版品，從內容呈現的轉換(Content Presentation)、元資料(Metadata)的轉換、閱讀編序與導覽(Sequencing and Navigation)的轉換和內容包裝(Content Packaging)的轉換等四個格式面向來開發此系統，期望將優質的數位學習內容(SCORM)也能以數位出版品(EPUB)的方式更加為使用者所利用與散布。

關鍵詞：SCORM, EPUB, 格式轉換, 數位教材, 數位出版品

The resources and applications about ebook have been changing the ways people read and get books, the rules and concepts of publication are significantly changed as well. Therefore, an important issue we have to face consequentially is how to rapidly produce valuable digital publications. In fact, there have been a lot of excellent e-learning contents produced and stored as SCORM format in repositories or learning management systems. But unlike the convenience of enjoying ebooks with a reader, these excellent learning contents have much more complicated design issues and have to be put on specific learning systems due to the requirements of learning objects' sharing, tracing, conformance, etc. In addition, these excellent learning contents were normally produced by highly qualified teachers who devote their professional knowledge to benefiting learners. We hope that these valuable contents can be more widely distributed and easily enjoyed as general ebooks for more people. As a result, we propose a converter which is able to convert SCORM compliant

learning contents to EPUB ebooks. The research details the convert process including the four modules; Presentation Transforming Module, Metadata Transforming Module, Sequencing and Navigation Transforming Module and Packaging Transforming Module. We look forward to seeing the converter facilitating the exchange and distribution of excellent digital content.

Keyword: SCORM, EPUB, Converter, Digital publication

壹、 前言

近來電子書資源與應用藉由電子閱讀器的普及，快速地席捲人們日常生活與閱讀習慣，也改變了傳統出版業的生態環境，數位閱讀趨勢嚴然已經成為一股不可抵擋的未來趨勢，政府也由新聞局、工業局等相關單位相繼投入資金研擬相關整合政策與配套措施，其中兩個重點就是在數位出版品的內容產出與如何提升數位閱讀的環境與風氣，在產生製作方面的最重要的國際電子書標準就是 EPUB。另一方面我們也關注數位學習的發展與成果，近幾年不論在台灣或國際上的數位學習應用也受到相當的重視，除了數位學習標準(SCORM)與系統的應用與發展之外，最重要的就是累積相當多的優秀數位學習內容與教材。

SCORM 的全名 Sharable Course Object Reference Model 起源於 1997 由美國白宮科技辦公室與國防部所共同推動的分散式學習主導計畫 Advanced Distributed Learning Initiative 中提出的一套標準，希望學習物件透過此標準建學習物件重複使用與共享的特性，幫助縮短 位教材的課程開發時間減少位教材的課程開發成本及位教材的課程能在各種不同的學習平台之間。EPUB 則是由國際數位出版聯盟 IDPF (International Digital Publishing Forum)所提出的一個數位出版品之交換格式與傳遞的公開標準。作為電子書內容描述的標準規範，EPUB 中定義了數位出版品呈現、封裝與編碼架構並加強 web 內容之語意。EPUB 能讓出版商在發行、傳遞與使用者使用時，在不同應用程式與閱讀載體間容易互通。這兩個標準的發展演進，**Error! Reference source not found.**為 EPUB 的歷史演進，IDPF 從 1999 年即發佈第一版電子書標準 OEBPS1.0，其主要是專為電子書的內容呈現及結構定義標準化，至到 2007 的大改版並改名為 EPUB2.0，此版本才開始對數位出版品的格式有了較全面性的標準規格，但其設計只適合簡單的文字版面，在中文的支援上仍有許多限制。例如文字書寫方式、閱讀方式、注音符號與圖文影音的支援性，都仍較單一且不足。最新的 EPUB3 為 2011 年所公佈，在多媒體與排版上有了極大改善。而 SCORM 自 2000 年的第一個版本以來，目前最新的版本為 2009 年所修正的 SCORM 2004 第四版，如圖 2 所示。



圖 1 EPUB 標準的歷史演進



圖 2 SCORM 標準的歷史演進

因此藉由本研究提出的數位學習教材與數位出版品轉換系統，我們所要努力的兩個目的：

1. 將優質數位學習教材轉換為優質數位出版品

目前在數位學習的發展與成果中有無數的優質數位學習教材與內容存在著，這些教材內容大多儲存在教材儲藏庫或學習平台中，由於在教材的設計與執行上必須搭配特定的學習環境與規範管理，因此在內容使用度上與能見度上都非常受限於平台的建置與策略的規範，但在去掉這些學習策略與追蹤管理機制等特性，回歸內容本質與閱讀行為的角度來看，這些內容本質其實都是非常優質且珍貴的數位內容。

2. 對內容產出者例如教師個人提供簡易快速的數位出版轉換工具

許多認真又熱心的教師花費大量時間與心力製做符合標準的數位內容與教材，可以說是集個人教學與研究之心血而成，現在透過本研究的轉換系統可以將這些教師累積下來的心血內容轉製成數位出版品，也因此卸下原有數位學習的設計包袱與門檻，將來在新增內容及擴充上都將更加容易，同時面臨未來個人出版的趨勢與環境下，對教師來說會是另一種型態的具體成果。

貳、 數位學習標準 SCORM 與電子書標準 EPUB 的比較

SCORM2004 規格書分為四個部分，如圖 5 所示：

1. SCORM 概觀(SCORM overview):

ADL 整個計畫的介紹及 SCORM 的技術規格概述。

2. 內容整合模型(Content Aggregation Model, CAM):

SCORM 課程物件內容整合模型旨在定義如何 別描述各種學習元件；學習元件應如何封裝 package 才能在 同的系統間交換；如何對學習元件進 描述，以方 元件之搜尋。

3. 執行環境(Run-Time Environment, RTE):

明學習平台如何啟動結束學習元件方法的 API 及學習元件如何與學習平台溝通以及溝通時記 學習者資訊的資 模式 data model，使設計者所製作的學習元件，可在學習平台中順利執 。

4. 編序和導覽(Sequencing and Navigation, SN):

說明如何編輯教材的學習順序

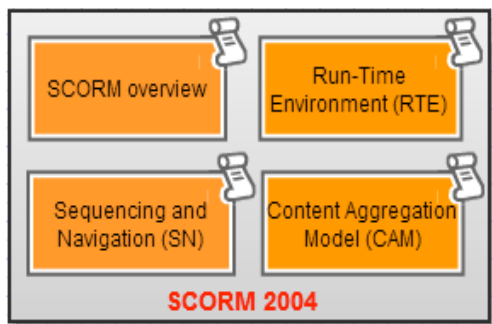


圖 3 SCORM 2004標準的內容組成

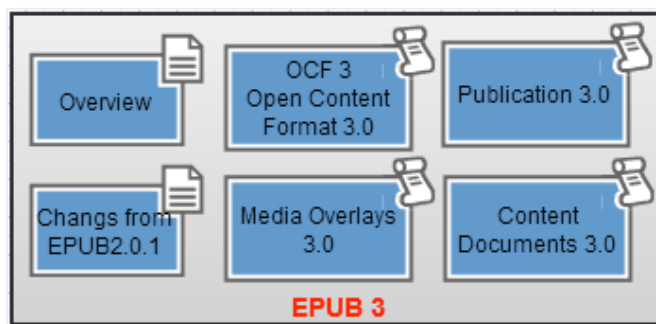


圖 4 EPUB 3標準的內容組成

另一方面現階段的 EPUB3 由主要 4 個標準所組成，如圖 4 所示：

1. **EPUB Publications 3.0:**
定義出版層級的語義和描述資料，以相容於出版標準的條件需求
2. **EPUB Content Documents 3.0:**
定義了使用 XHTML、SVG 和 CSS 在電子書內容中的使用方法
3. **EPUB Open Container Format 3.0:**
定義了如何將所有電子書中相關資源打包成一個電子書 EPUB 封裝格式
4. **EPUB Media Overlays 3.0:**
定義了文字與聲音的同步處理方式

在這邊值得注意的是 SCORM 標準相較於 EPUB 是非常龐大且複雜的，由於學習環境所提供的學習機制與資料管理扮演整數位學習的關鍵角色之一，因此 SCORM 主要還多包含了學習管理系統執行環境的標準(Runtime environment)，在本研究中主要是在檔案內容的轉換(由符合 SCORM 2004 課程包裝轉到 EPUB3 檔案格式)，因此應該是針對 SCORM 的內容整合模型，因此它才是說明如何包裝一個標準 SCORM 課程的規格說明書，所以以下的分析是針對 SCORM 內容整合模型標準與 EPUB 做比較，圖 5 為 SCORM 內容集合模型的內容架構。

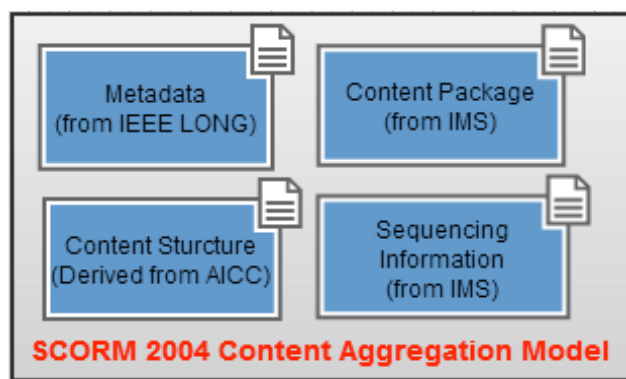


圖 5 SCORM 內容集合模型內容架構

由於我們在格式轉換上是數位課程與數位出版品的層級，因此比較對應的應該是

SCORM 的課程聚集模型(Content Aggregation)與 EPUB 本身，在此我們根據兩個標準的檔案格式內容區分為三大類來分析比較，如表 1 所示，此三類內容分別為(1) 內容呈現與資源型態 SCORM 的相關標準內容在 Content Model 章節而 EPUB 是 Content Documents 與 Media Overlays 標準；(2)元資料描述：SCORM 的相關標準內容在 Meta-data 章節而 EPUB 則是 Publication 標準；(3)檔案架構與包裝：SCORM 的相關標準內容在 Content Packaging 章節，而 EPUB 為 Open Container Format 標準；(4)編序與導覽規則：SCORM 的相關標準內容為 Sequencing & Navigation 章節，而 EPUB 則記載於 Content Documents 中。

表 1 SCORM 數位課程標準與 EPUB 數位出版品的內容比較

	SCORM 2004	EPUB 3
比較項目	Content Aggregation	EPUB 3
內容呈現與資源型態	Content Model	Content Documents 3.0 Media Overlays 3.0
後設資料描述	Meta-data	Publications 3.0
檔案架構與包裝	Content Packaging	Open Container Format (OCF) 3.0

一、 內容呈現與資源型態比較

SCORM 的內容單元是由 SCO(Shareable Content Object)所組成，而 SCO 則是由一個或一個以上 Asset 所組成，如圖 6 所示。



圖 6 SCORM 學習物件示意圖

對於素材 (asset) 的特性描述如下：(1) Asset 為學習資源之最基本形式；(2) Assets 是可被 web client 支援並呈現到學習者面前的電子媒體型，如文圖像聲音或任何資的部分；(3) 一個以上的 asset 集合可構成其他 assets；(4) Asset 可透過 meta-data 的描述，增進其在資源庫 repositories 中之搜尋及發掘，以提高再用 reuse 的可能

而 SCO 則是可分享的內容物件且是學習系統運做執行時的最小單位，其特性描述如下：(1)一個 SCO 是由一或多個 assets 組成；(2)為單一可被啟動的 launchable 學習資

源； (3)運用 SCORM RTE 與 LMS 溝通； (4)一個 SCO 即代表可被 LMS 追蹤的最低集合層級 lowest level of granularity 之學習物件 LMS 運用 SCORM RTE data model 追蹤 SCO； (5) 為提高再用性，SCO 應獨立於學習情境 context；也就是，SCO 要能在不同的學習經驗下被再用，以實現不同的學習目標； (6)Activity 可以集合(aggregate)一個以上的 SCO resource 或 asset resource，進而形成較高階(higher-level)的教學或訓練單元，或實現較高階的學習目標； (7)SCO 可藉由 meta-data 的描述，增進其在資源庫中之搜尋及發掘，以提高再用的可能。

每個 EPUB 出版品包含一份以上的 EPUB 內容文件，且這些內容文件可以是 XHTML 或 SVG 文件，以及描述出版品的可讀內容和其參考的相關資源（例如，影像、音訊和視訊片段）。EPUB 與 SCORM 在組成觀念上相同，出版品是由多個資源組成，再依預設的閱讀順序讓使用者透過閱讀系統進行閱讀導覽。而 EPUB 在內容呈現上採用 XHTML，它並不像 SCORM 在內容資源上強調學習物件(SCO)的重覆使用性，EPUB 在內容單元上主要是採用內容頁面為單位以 XHTML 或 SVG 為檔案格式，它目前主要的幾個特色如下：

1. 支援 HTML5 與 CSS3

語意上的應用雖然說是支援 HTML5，但這部分還在發展中，差別只是在支援程度上的差異，例如 scripting 和 forms 就沒有強制一定要支援。做為一個 XHTML 檔案格式，最主要的要求就是要它必須符合 XML 文件的規則限制以及它的語法，以產生合法的 XHTML。另一方面則是 CSS3 的支援，EPUB 在內容呈現的觀念是內容呈現應能適應使用者，而不是使用者必須適應內容，也就是說有動態調整排版以適合使用者閱讀裝置，透過 CSS3 允許多種不同語言和文化的排版。一些具體增強功能包括：(1)直書支援，也提供閱讀系統有能力讓使用者做方向切換；(2)更好的強調點處理；(3)更好的斷行控制(4)更好的斷字控制。

2. 可使用 Scripting

EPUB 支援允許在 HTML5 和 SVG 中的 scripting，這意謂著與書內容的互動可能性，但這並非必要條件，閱讀系統可以選擇性支援 script，例如在為了安全理由可將其停用，以避免危害使用者端載具也降低閱讀系統存取的障礙性，增加了 Scripting 由於沒有統一 API 介面因此也會降低存取性和內容再用性的可能。

3. 發音與媒體層

EPUB3 中採用 DAISY Digital Talking Books 功能，利用 HTML 針對動態版面配置，所設計的語意來做為發音的呈現準則，提供出版品文字和音訊同步的能力。

4. SVG 內容文件 (SVG Content Documents)

雖然 EPUB 出版品的頂層文件類型通常是採用 XHTML 內容文件，但其實也是可以採用 SVG 內容文件 (SVG Content Document)可縮放向量圖形，通常僅用於某些特殊環境，像是當最終形式(final-form)的頁面影像為該內容唯一合適的展現時，例如說在漫畫或連環圖畫書的環境下。

SCORM 與 EPUB 組成的資源類似但各有不同層次的特性與描述，如表 2 所示，SCORM 的內容單元是專注在 SCO 並強調可再用與分享，而 EPUB 是在 XHTML 則強調普遍相容的文件呈現格式。SCO 是需要和 LMS 溝通，所以有搭配的 API，新的 EPUB 在支援 HTML5 方面也有 scripting，但標準中並無與伺服器溝通的相關內容，主要是在使用者端的內容呈現或互動。

表 2 SCORM 與 EPUB 內容組成資源的特性比較

SCORM	EPUB
有學習物件的單位 SCO，強調可再用與分享特性	以 XHTML 格式為主，支援 HTML5，可包含各種素材的組成著重在檔案呈現使用 XHTML 格式與適應於各種載具上閱讀
SCO 可以是許多不同的 Asset 組成	在出版資源中與 Asset 類似，可使用的資源素材格式則定義在 Core Media Types 中
為單一可被啟動的 launchable 學習資源，透標準 API(script)與 LMS 溝通	無關伺服器端，內容呈現與使用只與使用者端與閱讀系統有關，支援的 Script 主要用在強化內容與使用者的互動效果
SCO 與 Asset 可有 meta-data 的描述	有對整本電子書的 metadata，沒有對單一書本章節專屬的 metadata

二、元資 之比較

為 讓學習者可以透過搜尋尋找可能的片段線 ，並產生自所需要解決問題的知 ，因此各種層次的學習物件 必須設定元資料，SCORM 所規範的元資料主要引用自 IEEE LOM 及 IMS 的規範，包括 asset、SCO 或是整合後的單一課程(aggregations) 可以賦予元資料，其類別如表 3 所示：

表 3 LOM 元數據的 9 個類別

類別	說明
一般(General)	描述描述整個學習對象的一般資訊
生命期(Lifecycle)	描述與該學習對象的歷史和當前狀態有關的特性，以及在其演化過程中發揮作用的那些特性
元-元數據(Meta-Metadata)	描述關於元數據實例自身的信息（而不是元數據實例所描述的學習對象）。
技術(Technical)	描述學習對象的技術需求和技術特徵
教育(Educational)	描述學習對象的教育性和教育學特徵
版權(Rights)	描述使用學習對象的知識產權和條件
關係(Relation)	範疇包括定義學習對象和其他相關學習對象之間的關係的特性
註解(Annotation)	提供學習對象的教育性應用的註釋，並提供何時何人創建該註釋的信息

分類(Classification)	描述該學習對象和特定分類系統的關係
--------------------	-------------------

EPUB 出版品則提供許多豐富的選項以新增出版品的書籍資訊，套裝文件包括有關出版品專屬書籍資料部分，能容易取得出版品的名稱、作者、識別符和其他資訊，利用 link 元件，將其他書籍相關資訊附加到出版品的方法。利用 unique-identifier 屬性，建立用於出版品的唯一識別符，結合出版品的最後修改日期，可定義出套裝文件識別符，用來提供區別同一個出版成品的不同版本。

具有此電子書的書目性且結構化的元資料如表 4 所示，最主要提供描述該如何處理且呈現此出版物，由一系列的標籤元素詳細說明此出版物的出版級資訊，另外也說明每一個組成書的資源和閱讀順序。

表 4 EPUB 文件包裝元資料元素表

元素標籤名	說明
<package>	package 元素為此出版品包裝文件的最外層根元素，它封裝了出版相關的 metadata 和所用資源的相關資訊
<metadata>	此 metadata 元素封裝出版相關的資訊，它包含以下的 DCMES(Dublin Core Metadata Element Set)的元素如下：
<dc:identifier> (metadata 子元素)	這個 DCMES 元素記載此出版品一個唯一的識別碼，例如可以是 UUID, DOI, ISBN or ISSN。
<dc: title> (metadata 子元素)	說明此出版品的書名
<dc: language> (metadata 子元素)	說明此出版品採用的語系為何
<dc: data>	用來定義此出版品的出版日期，此日期並非最後修改時間，最後修改時間則是使用 modified 屬性
<meta>	用來加上額外資訊，再次說明描述或精煉原本的 metadata，閱讀系統若是不支援應忽略此元素。
<link>	用來組織此出版品的資源，說明資源的所在何處與屬性
<manifest>	定義書籍的所有元件、內含書籍的所有章節、頁面、內容。如文件、CSS、圖片等
<item>	描述書籍清單型態，該項目可能是文件、圖檔、工作表等
<spine>	定義書籍頁面的結構順序，依照 item 中參考順序
<itemref>	spine 的子元素，照清單定義項目。項目 考的順序，為 eBook 資源被閱 的順序

<guide>	出版品的功能性架構組件，輔助閱 系統能方 存取
<bindings>	對不支援的媒體類型，此元素允許定義了一系列系使用者特定處理的方法

由於 EPUB 在元資料的要求上雖然較單純，但由於是需要出版層級的元資料，因此在轉換過程中最重要的元資料來源將會是 SCORM 元資料中的一般（General）、元-元數據（Meta-Metadata）、以及版權（Rights）。

SCORM 元資料從 asset 到 SCO 再到整個課程都有 metadata，所以要填的地方很多，加上種類對多元，因此若要填較完整資料時會是一件相當耗時費力的工程，但也沒有強制一定要全部填滿這些資訊，EPUB 就是針對此出版品在出版層級所需的元資料所以單純的多，以下表 5 為兩者間的比較。

表 5 SCORM 與 EPUB 的元資料特色比較

SCORM	EPUB
繼承 IEEE 與 IMS 的對學習物件的後設資料標準	原生定義出版相關的元資料
從 Asset、SCO 到整個學習課程都可以有後設資料	主要在描述出版品層級的元資料
可填的後設資料量相當大，且種類複雜多元，不一定要全部填	後設資料只有針對出版品本身，因此量較少，種類單純

三、學習與閱讀導覽順序(Sequencing and Navigation)

SCORM 編序與導覽(SN)是以 IMS 的精簡編序(SS)規格為基礎。IMS 精簡編序(SS)定義了呈現預定學習行為的方法，而學習管理系統可以依此將不同的學習活動用一致的方法加以編序。另外，SCORM 編序與導覽(SN)亦定義了在 SCORM 的環境裡如何應用與延伸 IMS 精簡編序(SS) 規格，並基於學習者在開啟內容物件與編撰的編序策略時的互動行為結果，描述了學習行為在學習活動樹(Activity Tree)的分支與流程。

SCORM 編序與導覽(SN)共包含三大模組，分述如下：

1. 編序定義模型(Sequencing Definition Model):

闡明如何在開發內容時運用的描述編序策略的編序資訊的每一個環節，與使用範例。

2. 編序行為(Sequencing Behaviors):

詳述為了編序的目的與追蹤資訊(tracking information)的細節，與學習者進程的內容物件如何作用於追蹤資訊。

3. 導覽模組(Navigation Model):

描述在編序階段，使內容物件能夠查詢學習管理系統與標示學習管理系統所需的導覽資訊需求的執行資料模型。提供對學習管理系統對學習者提供適當的導覽控

制的指引。

轉換成符合 SCORM 2004 之語法, 在 SCORM 2004 規範中則是提供了 8 大類語法分類如下:

1. **Control Mode:** 設定教材章節之流程
2. **Sequencing Rules:** 設定各個教材章節之條件(Condition)及行為(Action)
3. **Rollup Rules:** 描述子章節與父章節之學習狀態之間的關係
4. **Auxiliary Resource:** 教材輔助教材
5. **Objectives:** 設定章節之學習目標
6. **Limit Conditions:** 設計章節之閱讀限制條件
7. **Randomization Controls:** 設定教材章節是否隨機出現
8. **Delivery Controls:** 設定是否記錄章節之學習狀態

在 EPUB 中的導覽閱讀順序機制允許作者來制定一個人機皆可了解的導覽資訊, 它是使用 HTML5 nav 元素標籤來定義人機可讀的導覽資料, 在舊版的 EPUB 是使用 NCX 文件, 對 HTML5 文件本身, 導覽文件也提供使用 CSS 和 hidden 屬性來調整顯示的彈性, 而不會影響閱讀系統的資料存取。

每個 nav 元素可能包含一個可選的標題, 說明標題導航列表。標題必須 H1 到 H6 元素, 而接下去要接一個 ol(ordered list)為有序的列表, 再接下去為 li(list item), li 後就是目標內容的實際位置, 但也可以是 span 標籤做為描述列一個標題所用, 比較內容如表 6 所示。

表 6 SCORM 與 EPUB 的編序導覽特色比較

SCORM	EPUB
<ul style="list-style-type: none">● 8 大類學習順序的規則與語法● 設定相當複雜, 製作門檻高● 學習策略變化多、彈性高	<ul style="list-style-type: none">● 以線性閱讀為主● 設定較簡單, 製作門檻低● 導覽策略單調、彈性低

SCORM 的編序與導覽比 EPUB 複雜很多, 由於教育的學習策略與學習路徑是相當多元且富有理論性, 加上要讓教材設計者有彈性地設計學習順序, 因此在規則與策略上的設定就相當複雜也造成設計者在教材設計上的難度。反觀 EPUB 的導覽, 以傳統書籍的閱讀方式來說通常是以線閱讀為主, 例如小說、故事書...等, 但書也有非線性類的內容, 例如食譜, 參考書...等, 雖然書本的閱讀順序沒有特定限制, 但與學習順序策略相比是相對簡易的。在這方面的轉換對應是由於是行為的控制資訊, 因此在 SCORM 的編序與導覽內容中是由 sequencing rule 所負責。

四、 內容包裝(Content Packaging)

SCORM Content Packages 之目的為: (1)作為一標準化的方式, 讓學習內容能在不同的系統或工具間交換; (2)提供用來描述結構或組織, 以及一組學習內容之預定行為 (intended behavior) 之方法; (3)產生能在學習管理系統、開發工具、與內容資源庫間互

通的課程包裹。圖 7 為課程包裹的結構圖。

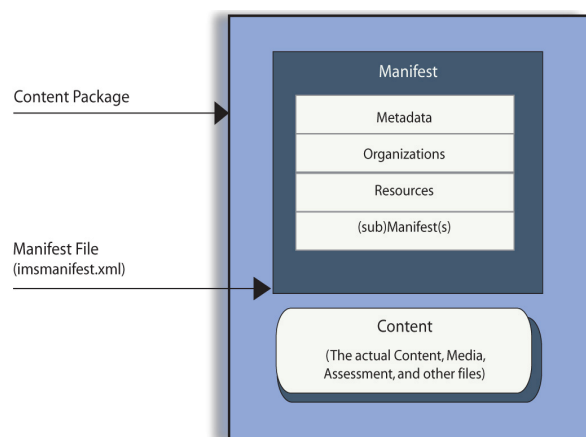


圖 7 SCORM 課程包裝結構

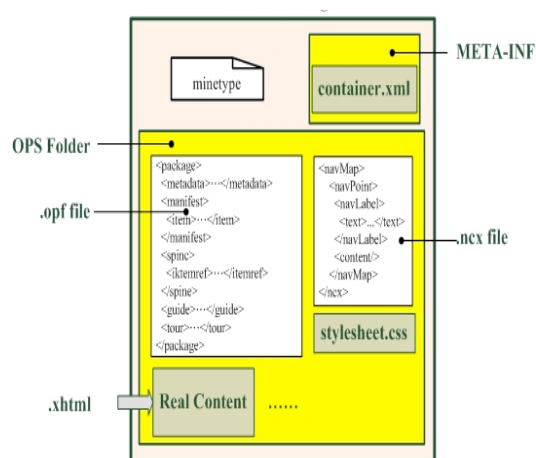


圖 8 EPUB 出版品包裝結構

EPUB 開放容器格式(Open Container Format, OCF)定義一種檔案格式及處理模型，將構成一個以上 EPUB 出版品的相關資源，封裝於單一檔案的容器之中在不同的個人或平台間進行交換的容器格式，如圖 8 所示。OCF 規範以抽象的方式定義構成檔案集合的規則：「抽象容器」。同時也定義此抽象容器在 ZIP 壓縮檔裡的呈現規則：「實體容器」。實體容器是建構在所使用的 ZIP 技術上，兩個標準的包裝結構比較如表 7 所示。

表 7 SCORM 和 EPUB 包裝結構差異

SCORM 的課程包裹組織	EPUB 數位出版品包裝組織
元資料(Metadata): 描述這個課程物件的特徵或屬性	META-INF: META-INF 的目錄之中，必須包括名為 container.xml 的檔案。在此 container.xml 檔之中，必須指明容器所包含的 EPUB 出版品根檔案其媒體型別與路徑
組織結構(Organizations): 描述這個課程物件的組織與結構	.OPF file: 描述這個數位出版品的內容結構
資源(Resources): 指出在這個課程物件中所需要用到的實體資源檔案的位置	Manifest: 指明容器所包含的 EPUB 出版品根檔案其媒體型別與路徑
子元資料(Sub-Manifest): 描述這個課程物件的組成及順序規則	Spin: 描述這個出版品的頁面內容順序
實體檔案(Physical File): 包含於該課程範疇的學習內容檔案	實體檔案: 包含該數位出版品內容檔案

參、 系統模組實作測試

本研究的實作模組分為四個部分：1)內容呈現轉換模組(Presentation Transforming Module)，2)元資料轉換模組(Metadata Transforming Module)，3)編序與導覽轉換模組(Sequencing Transforming Module)，4)包裝與檔案結構轉換模組(Packaging Transforming Module)，詳細說明將分述如下：

一、 內容呈現專換模組(Presentation Transforming Module)

此模組主要有兩個部分的呈現工作要完成，第一部分為將課程的導覽樹(Navigation Tree)轉為書目結構(Table of Content)，第二部分為將課程的內容檔轉為電子書內容呈現檔。

(一) 書目格式的產生

一個 SCORM 課程的樹狀結構圖是可以由 manifest 檔案中的 organizations 標籤中解析產生的，在 organizations 中的基本定義就類似是此課程的錄目一樣，如圖 9 左半邊的 SCORM manifest.xml 所示，藉由此標籤中對課程章節的排列與組織來產生一個樹狀且有層次的導覽結構，通常在樹狀節構中，每一個節點代表一個 HTML 檔案也代表一個章節，還有包含學習順序的資訊(Learning Sequencing)，此部分的資訊將在「編序與導覽轉換模組」中進行處理。圖 9 的右半邊則顯示了對應 EPUB 的書目資訊位於 package.opf 中的區塊位置及 TOC 目錄檔中的區塊位置。

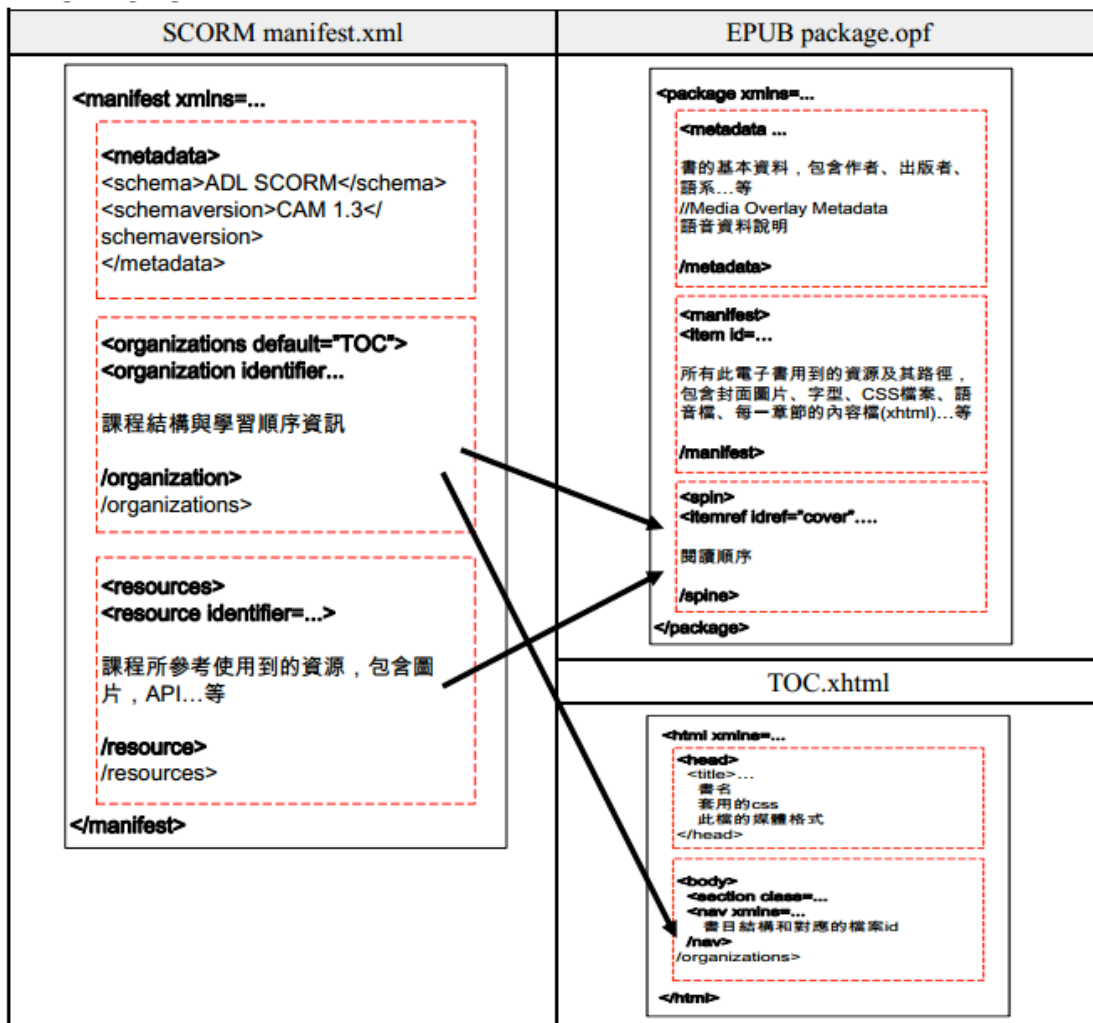


圖 9 SCORM 與 EPUB 的目錄資訊對應區塊圖

EPUB 的書本目錄資料儲存在 opf 檔的 spine 標籤下與 toc.xhtml 檔案中，主要有兩部分：一開始描述書的題目和通用 CSS 檔案，第二部分即是 EPUB 中的書目格式描述資料存在<section>標籤中，詳細的瀏覽順序則在<nav>標籤下以來組織結構，最後顯示的每個章節名稱還要有超連結<a>在記載，方便讀者使用超連結點選要閱讀的章節。由於書本的書目結構較單純且主要是以線性閱讀，在書目錄中的資訊主要只記載對應的內容檔案名稱而已。

因此主要的方法原則就是在 SCORM manifest 檔案中的 Content Organization 中的<item>與<title>將要對應到 EPUB TOC 檔案中的與標籤，然後在根據對應到的 Content Aggregation 裡的真正學習資源的檔案路徑找出其實體路徑，並轉換成 TOC 檔中的<a>標籤內容。

(二) 內容檔案的呈現

此步驟是將呈現內容的 SCO 轉為 EPUB 可讀的 XHTML 文件，由於 SCO 可能是各種 Asset 所組成，但大部分是以 HTML 為呈現的學習內容，學習內容 SCO 只要是 HTML 頁面則可能是 HTML4 是屬於散鬆語法內容，因此這個步驟重點就是在於從 HTML 轉

換為 XHTML 的過程轉換步驟如表 8:

表 8 HTML 轉換為 XHTML 步驟表

執行步驟	動作說明
步驟一 轉換文檔為良好結構	所謂良好結構 (Well-formed) 是由 XML 引入的一個新概念。也就是說所有的元素都必須有結束標籤或者以特殊的方式書寫，而且所有的標籤必須合理地嵌套。儘管如此，交叉使用仍然是合法的，而且在現有的瀏覽器中也能夠被廣泛接受。
步驟二 元素名稱和屬性改為小寫	XHTML 文件要求所有的 HTML 元素名稱和屬性名稱都要小寫。因為 XML 本身大小寫意義不同，因此必須區分開來。比如，和是完全不同的。
步驟三 將非空標籤，加上結束標籤	基於 SGML 的 HTML 4 裡面，允許特定的標籤省略結束標籤；這些元素暗含有結束標記。XHTML 不允許省略結束標記。所有元素 (包括在 DTD 中聲明為空的標籤)，都必須有結束標籤。在 DTD 中聲明為空的元素可以用結束標籤或者使用空元素速記法。
步驟四 屬性值必須總是使用引號包裹：	所有的屬性值都必須使用引號包含，包括那些以數值類型出現的。
步驟五 檢查禁止屬性簡化：	XML 不支持屬性簡化，屬性值對必須書寫完整。屬性名，像 compact 和 checked 在沒有指定具體值的情況下，不能夠使用。
步驟六 將空元素加入結束標籤：	空元素必須有一個結束標籤，或者用 /> 來結束開始標籤。例如， 或者 <hr></hr> 。

二、 Metadata 轉換模組 (Metadata Transforming Module)

圖 10 為 SCORM 數位課程中的 manifest.xml 元資料說明內容區塊與 EPUB 電子書中的 OPF 檔案中元資料的對應區塊位置，從圖中可以發現電子書中最重要的一個元資料說明檔。

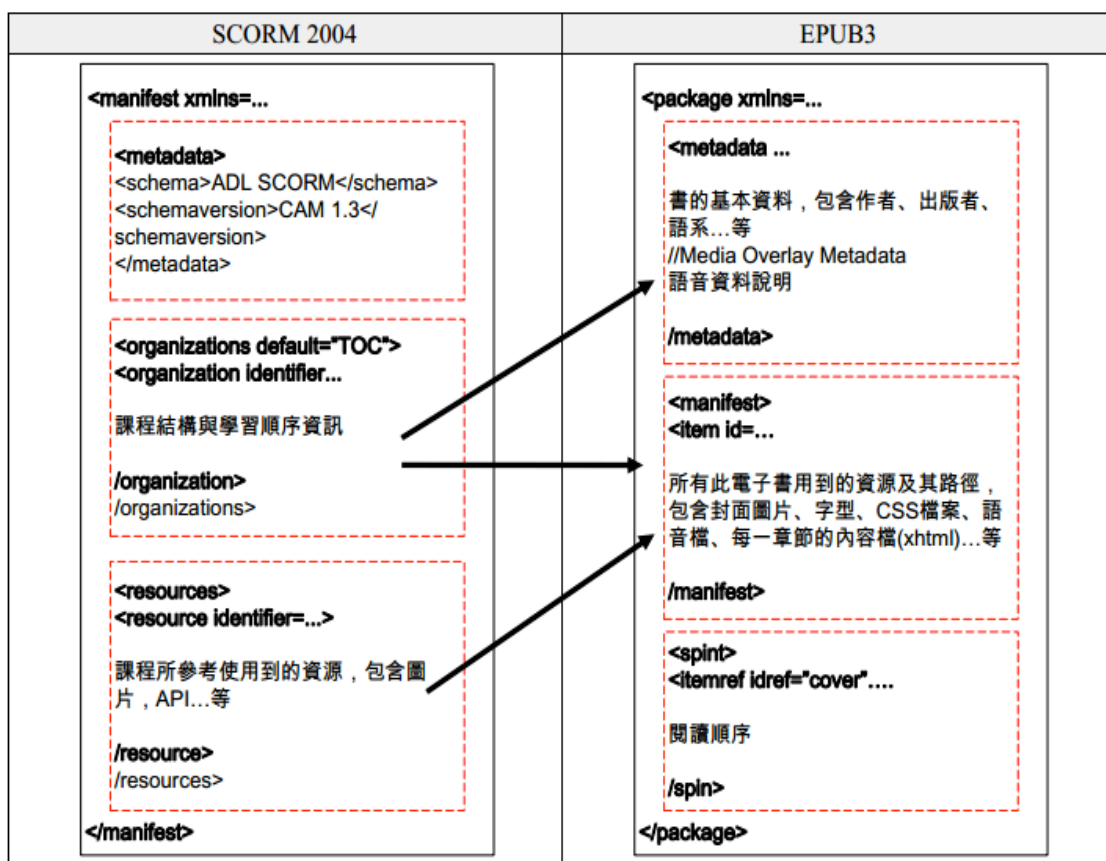


圖 10 SCORM 與 EPUB 的元資料對應結構比較圖

這裡分為兩部分；自動選填與手動填入，由於學習課程的 metadata 通常是關於學習物件本身與學習策略的描述，因此在轉為 EPUB 出版品時在性質上有不同，除了基本描述外，因為必需要有出版層級的資料描述，因此要提供給使用者自行輸入的介面以完整電子書的出版相關 metadata 的內容之描述，如標題、版本、使用語言或資料來源…等屬性，步驟如下：

1. 課程一般資訊轉為出版品相關資訊：

從 Content Organization 中找出數位課程的一般資訊轉為 EPUB 的基本資訊如 title 等，因為 EPUB 方面，其中包含有<dc:title id="title">書名、<dc:creator id="creator">作者名、<dc:identifier id="pub-id">書的 id，例如一個 ISBN 碼、<dc:language>語系…等，其中書名、編號、語系是必填，需有輸入介面

2. 手動輸入

產生輸入介面讓使用者判斷是否要另加 EPUB 的其他元資料原本數位課程與出版品的元資料的著重項目不同，因此要有讓使用者輸入的介面資源相關元資料轉為出版品的資源元資料

3. 將 Content Aggregation 中資源列表轉填至 EPUB manifest 標籤下

將 SCORM imanifest.xml 中<resource>標籤下的<file>有包含資源檔案名稱與位置轉為相對應 EPUB package.opf 中<manifest>標籤下的 item。

三、 包裝轉換模組(Packaging Transforming Module)

包裝轉換模組的工作在於最後將數位出版品打包依照開放式內容格式(OPF)，在這個模組中最主要的工作其實是產生 EPUB 出版品包裝中必要的資訊檔案為了給閱讀系統相容及讀取等需求，因此兩個主要動作，分別為產生 EPUB 出版內容必要的指引與媒體類型說明檔，另一個則是將其他模組結果填入相對應的檔案中，它們兩者的包裝格式如圖 11，詳細步驟如下：

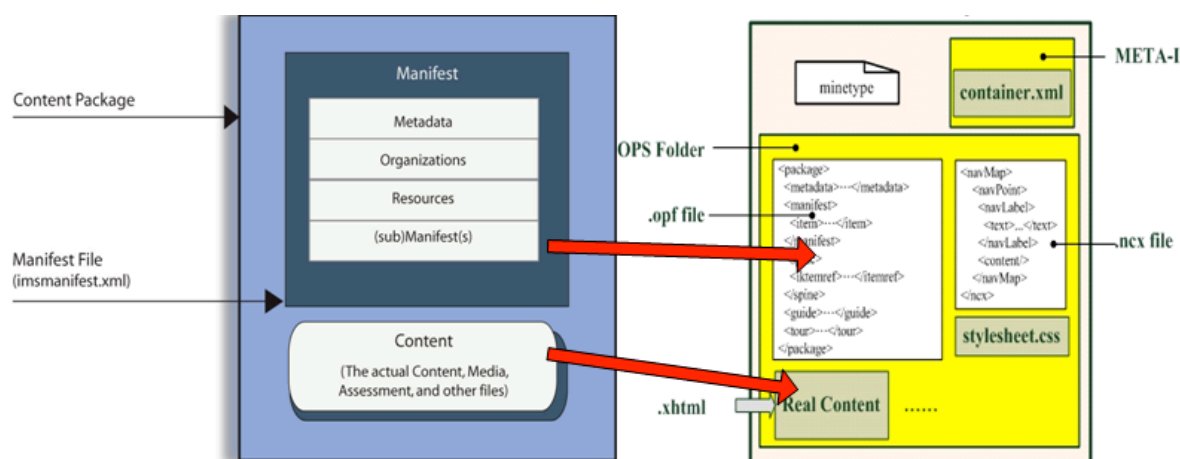


圖 11 SCORM 與 EPUB 內容包裝結構對應圖

轉換步驟：

1. 產生 minetype 檔案

Minetype 內容說明此數位出版品的媒體格式為何，對應使用者的閱讀系統進行開啟與解析，標準的內容一般為 application/epub+zip。

2. 產生 META-INF 目錄夾

產生 META-INF 目錄夾中的 container.xml 於 META-INF 目錄中，此檔案將引導閱讀器指向正確的 opf 檔案為置，並說明此 opf 的檔案媒體類型。

3. 產生 OEBPS 目錄夾

產生 OEBPS 目錄夾中的.opf 檔，其內容將由內容呈現模組與後設資料模組的產出結果寫入產生。包含出版層級的後設資料，manifest 資訊和閱讀順序等。

4. 將所有資源檔案放置於 OPS 目錄下

所有數位課程資源 Asset 與 SCO 都存至 OPS 下，並轉為 XHTML 與一般出版品的允許資源型態。

5. 封裝所有內容為 zip 檔格式，再更改副檔名為 epub

肆、 系統實作與測試

由於 SCORM 課程包中的課程資料，以及 EPUB 數位出版品中的資料內容，大多數皆由網頁語言 HTML5 所編輯而成。因此，為提供一項服務，能讓使用者上傳 SCORM 課程包至 website 以後，經由後端的 server 將 SCORM 課程包轉換為 EPUB 數位出版品，

讓使用者下載轉換後的 EPUB 數位出版品，並且載入 EPUB 閱讀器中能正確的觀看，盡可能的讓兩者的資料透過簡單快速的方式溝通與轉換。

表 9 系統測試環境說明表

系統平台類型與開發工具	轉換內容的來源	轉換結果測試工具
1. 以網頁為基礎的系統平台 2. PHP、HTML5、DOM	SCORM 2004 4th Edition標準課程範例	Radium ; IDPF 支持的 EPUB3閱讀軟體 (Chrome 擴充元件)

表 9 為此次測試環境說明表，本研究使用了 PHP 與 HTML5 的網頁程式語言作為系統開發的基礎，並依據 Document Object Model (DOM)的標準作為轉換檔案的主軸。實驗中使用了一台網站伺服器，以及 SCORM 2004 4th Edition 網站中所提供的標準課程包，來做為實驗的設備以及素材，並將數位出版品依照 EPUB 3.0 的規範作轉換。

最後測試平台 Radium 則是一個 EPUB 出版品閱讀軟體，它是由 IDPF 所支持的提供一個開放源始碼的 Chrome 擴充元件，可閱讀 EPUB3.0 檔案也下相容 EPUB2，同時內建檢驗器(Validator)，符合最新 EPUB3 以網頁為基礎支援 HTML5、CSS、JavaScript、SVG 和 DOM。

一、系統操作與執行流程

圖 12 為此系統在將 SCORM 課程包轉換成 EPUB 出版品的詳細流程圖。首先會先讓使用者上傳欲轉換之 SCORM 課程包，當系統接受了使用者所上傳的 SCORM 課程包以後，將會對課程包內的檔案 (imsmanifest.xml) 作徹底解析，並從其中萃取稍後轉換時所需要的資訊。

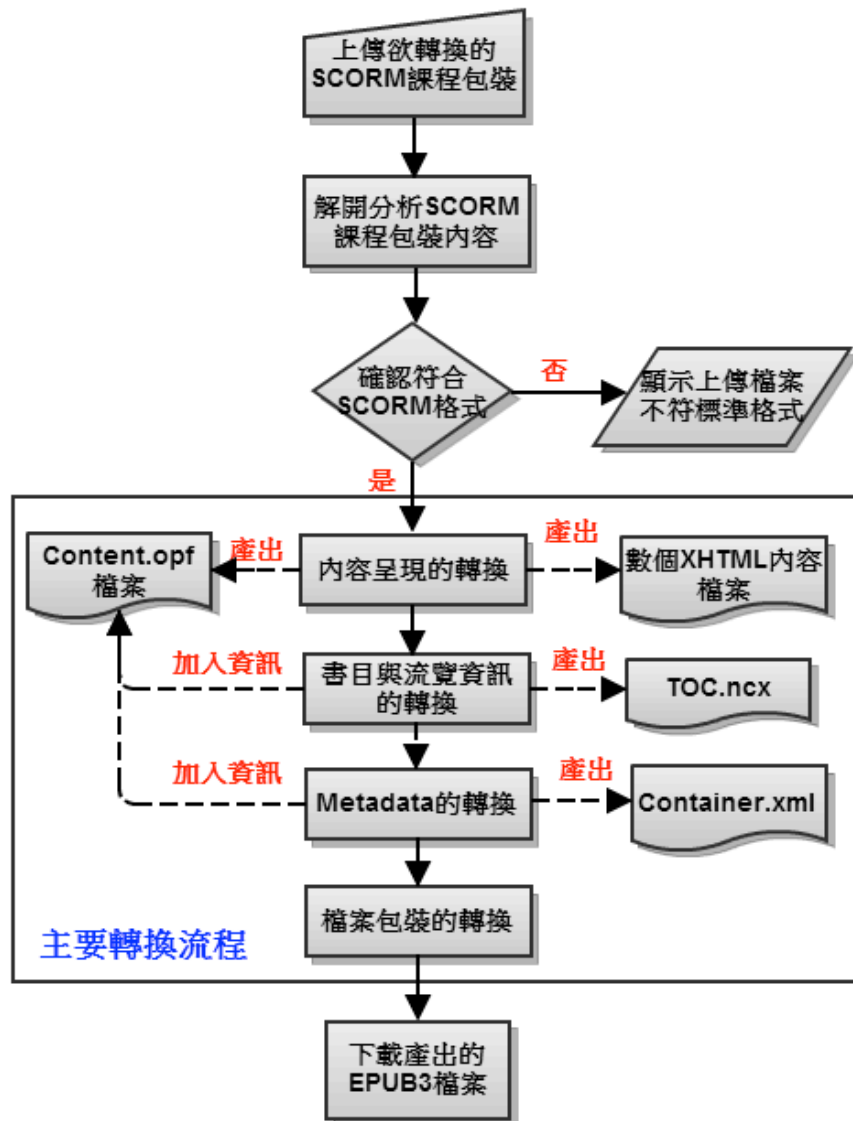


圖 12 系統執行流程圖

主要的轉換流程為：

分析課程包組織並產生數位出版品的架構，內容呈現的轉換部分將 SCORM 教材內容轉為 EPUB 出版品的書頁內容，以 XHTML 格式儲存並將所使用到的資源檔案寫入 content.opf 檔案中。

產生的流覽資訊將分別寫入 content.opf 的元資料中(<spine>)，而整個書目結構則產出為 TOC.ncx 檔案。

Metadata 的轉換資訊也是寫入 content.opf 中，並產出 container.xml 儲存於 META-INF 目錄中，此檔案將引導閱讀器指向正確的 opf 路徑資訊，並說明 opf 的檔案媒體類型。

最後轉換完成的檔案經過整合封裝後成為標準的 EPUB 格式，直接讓使用者下載

二、系統操作介面

首先，本系統將讓使用者透過網站上傳 SCORM 的課程，如圖 13 所示。在本實驗當中，本實驗將利用 SCORM 2004 4th Edition 的網站所提供之範例課程包做測試。接著，在選定 SCORM 課程包上傳成功後，稍待一會兒就會看到轉換流程已經完成的訊息。

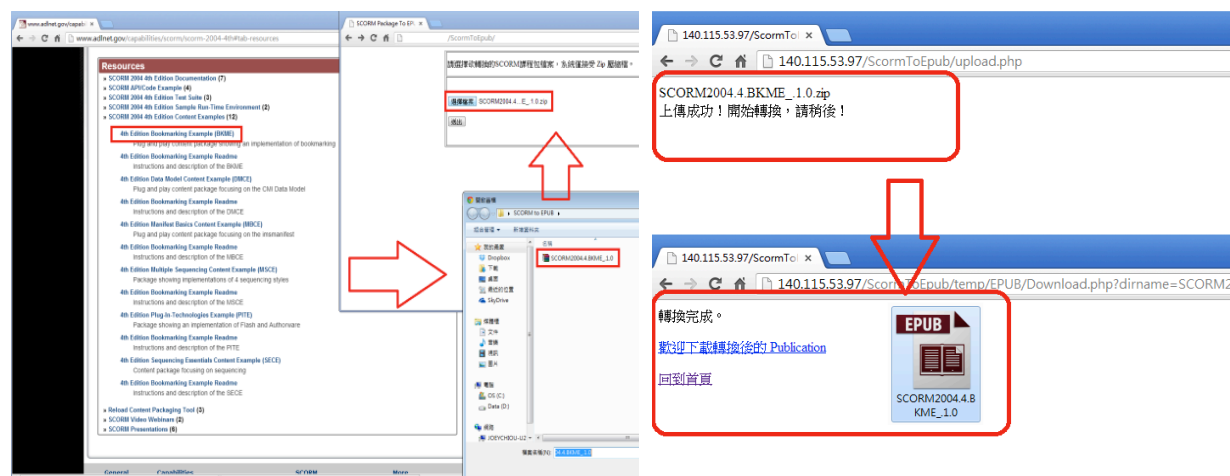


圖 13 系統操作介面與結果下載

最後當使用者點選完成訊息中的下載連結，可將轉換後的數位出版品下載至個人電腦或是其他載具。當使用者將下載的數位出版品匯入 EPUB 閱讀器中，即可由 EPUB 閱讀器正確的觀看轉換後的 SCORM 課程之數位出版品，圖 14 為轉換後的 SCORM 教材在 EPUB 閱讀軟體 Radium 中觀看的情形。

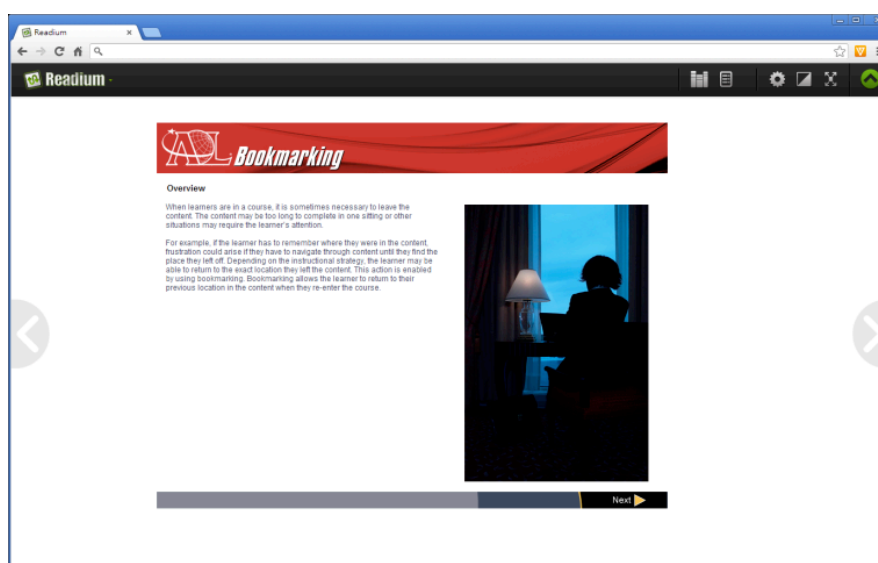


圖 14 轉換後的數位出版品在 EPUB 閱讀器中開啟

伍、 結論與問題討論

EPUB3 做為一個現代化的數位出版品標準提供了在數位閱讀時代中，滿足讀者在多媒體與互動功能上的各種需求，也為數位出版品在新時代的外貌輪廓做了明確的呈現與定義，但在內容的生成與規劃上依然要仰賴人的創造力與時間經驗，其中所花費的成本經常是令人怯步的原因，因此本研究的内容格式轉換系統的貢獻可以分為三個層面：1) 對一般使用者來說，在於能夠將深值教育價值的 SCORM 數位教材，跳脫嚴謹限制的學習策略框架與收集數據資料的任務需求，成為使用者能夠單純享受閱讀樂趣的數位出版品。2) 對教材製作者而言，這些人可能大部分也是學有專長的教師或教育從業人員，對他們付諸心力而成教材內容，將可以以電子書的方式在個人出版的機會上，做為個人在專業領域上成果的具體展現。最後3) 對整個數位出版環境上，優質的內容是吸引讀者的主要原因，而靈活快速的轉換功能將降低產出成本，更利於後續出版專業的後製工作。

本研究的轉換的過程中，資料的不對等確實會造成對原內容的資訊遺失與新內容的資訊遺漏，例如數位教材著重於教育方面的元資料，描述學習的對象、目的、使用的教學技術…等，而出版品則以出版資訊為重，例如作者、語系、出版商…等，當然這些資訊可以留至後續專業出版人員編製，但未來在轉換過程中，也可再提供手動輸入介面，補足無法自動填入的元資料部分。而在內容呈現的問題上，所幸 EPUB3 宣布支援 HTML5，也就是說幾乎所有教材的多媒體內容與互動功能都能順利執行與呈現，遠優於 EPUB2，但可惜的是目前許多以 Flash 格式所製作的教材恐怕無法在 EPUB3 閱讀軟體上順利呈現。另外本研究的轉換目前是單向的，也就是從數位教材轉換為數位出版品，但無法逆向轉回，主要原因就是因為 SCORM 的執行環境與教材設計資訊相對龐大且複雜，導致在轉換過程中會遺失教材的大部分元資料與學習管理平台的溝通機制，因此在未來反向轉換的挑戰上，則是如何利用介面的設計，減輕必要的手動輸入資訊所造成的負擔。

誌謝

本研究感謝國科會編號 NSC101-2221-E-032-064 計畫支持研究經費。

參考文獻

1. ADL, "Sharable Content Object Reference Model (SCORM) 2004," vol. 2004, 2004 ed: <http://www.adlnet.gov/capabilities/scorm/scorm-2004-4th>
2. IMS (Instruction Management System) Global Learning Consortium , <http://www.imspjct.org/>
3. EPUB 3. International Digital Publishing Forum | Trade and Standards Organization for the Digital Publishing Industry. <http://idpf.org/epub/30>
4. Shih, Timothy K., Wang, Te-Hua, Chang, Chih-Yung, Kao, & Tai-Chien, Hamilton, Douglas (2007). Ubiquitous e-Learning With Multimodal Multimedia Devices. IEEE Transactions on Multimedia, 9(3), 487-499
5. Lin, Freya H., Shih, Timothy K., Kim, Won(2009). An Implementation of the CORDRA Architecture Enhanced for Systematic Reuse of Learning Objects, IEEE Transactions on

Knowledge and Data Engineering, 21(6), 569-577

6. Lin, Freya H., & Shih, Timothy K.(2008). Automatic Trap Detection: A Debugging Mechanism for Abnormal Specification in the IMS Sequencing Controls, IEEE Transactions on Learning Technologies, 1(3), 176-189
7. Yen, Neil Y. ; Shih, Timothy K. ; Chao, Louis R.,& Jin, Qun (2010). Ranking Metrics and Search Guidance for Learning Object Repository, IEEE Transactions on Learning Technologies, 3(3), 250-264
8. Marinai, Simone, Marino, Emanuele & Soda, Giovanni.(2011). Conversion of PDF Books in ePub Format, Proceedings of International Conference on Document Analysis and Recognition (pp.478-482)
9. Okuda, Shigeto ; Emi, Keiji ; Kawachi, Yusuke(2012), Building of an education system with electronic textbooks of the ePub format and with smartphones, Proceedings of Consumer Electronics (pp.325-328)