

私立東海大學
資訊工程與科學所
碩士論文

指導教授：朱正忠 教授

內文感知行動學習教材呈現之研究
Context-Sensitive Content Representation for
Mobile Learning

研究生：林弘信

中華民國九十四年七月

中文摘要

行動學習泛指學習內容可在任何時間、任何地點以及任何種類的設備上作呈現。現今，大多數的學習內容管理系統（Learning Content Management Systems，LCMS）已發展出便利的教材編輯工具（Authoring Tool），以協助授課老師製作上課用的學習內容，其中包括靜態文件，例如 Powerpoint、Word、PDF 與動態多媒體文件，例如 Video、Audio，然後整合這些學習內容以提供學習者最佳化的內容呈現。然而，大部分學習內容管理系統之呈現環境卻還是以桌上型裝置為主，例如桌上型 PC 或 Notebook，而非針對行動裝置，例如手機或 PDA。所謂的內文感知（Context Sensitivity）是指系統本身有能力偵測並且分析來自不同的設備內文。在本研究中，我們提出了一個內文感知中介軟體（Context-Sensitive Middleware，CSM），可將一份原本只能呈現在桌上型裝置的教材，依據行動裝置的不同作最適化的呈現，以達到行動學習的目的。

關鍵詞：行動學習、學習內容管理系統、內文感知

Abstract

Mobile learning means that the learning contents can be displayed anytime, anywhere, and with any kind of presenting device. Nowadays, Learning Content Management Systems (LCMSs) usually provide convenient authoring tools to help instructors to construct their learning contents, which may include static document such as powerpoint, word, pdf document and dynamic multimedia document such as video and audio files, and then integrate these learning contents to provide learners with proper contents rendering through access devices. However, most of LCMSs are based on desktop computer environments, rather than mobile devices. Context-Sensitivity is an application of software system's ability to sense and analyze context from various sources. In this paper, we develop a Context-Sensitive Middleware (CSM) for LCMS to transform the same learning contents to different mobile devices, so mobile learning can be supported.

Keywords: Mobile Learning, Learning Content Management System (LCMS), Context-Sensitive

目錄

目錄.....	I
圖目錄	II
表目錄	IV
第 1 章 緒論.....	5
1.1 前言.....	5
1.2 研究動機.....	6
1.3 研究目的.....	6
1.4 章節安排.....	7
第 2 章 相關研究	8
2.1 E-LEARNING 背景知識.....	8
2.2 CONTEXT-SENSITIVE 機制.....	19
2.3 XML & XSL.....	22
2.4 DESIGN PATTERN MVC.....	23
第 3 章 內文感知行動學習環境	25
3.1 系統架構.....	26
3.2 靜態文件解析.....	29
3.3 靜態文件存取.....	34
3.4 行動學習教材解析.....	37
3.5 內文感知行動學習教材建置.....	41
第 4 章 研究成果	47
4.1 實驗平台.....	47
4.2 實驗結果.....	48
第 5 章 結論與未來方向	53
參考文獻	55

圖目錄

圖 1：教材描述概念圖	11
圖 2：ULF模組關係圖	14
圖 3：LMS示意圖	16
圖 4：CMS示意圖	18
圖 5：LCMS示意圖	19
圖 6：MVC PATTERN 概念圖	24
圖 7：內文感知行動學習環境之系統架構圖	26
圖 8：靜態文件解析架構圖	30
圖 9：元素相依關係示意圖	31
圖 10：靜態文件版面佈置種類	32
圖 11：(A) 雙欄樣式 (B) 多欄樣式	33
圖 12：影像編號示意圖	33
圖 13：文字元素判斷示意圖	34
圖 14：靜態文件存取架構圖	34
圖 15：記憶體不足示意圖	35
圖 16：網頁切割示意圖	35
圖 17：雙向捲軸之瀏覽示意圖	36
圖 18：行動學習教材解析架構圖	37
圖 19：MANIFEST之檔案範例 (IMSMANIFEST.XML)	38
圖 20：每一頁投影片之影像解析示意圖	39
圖 21：投影片內容之影像解析示意圖	39
圖 22：投影片之大綱文字解析示意圖	40
圖 23：投影片之註解文字解析示意圖	40
圖 24：多媒體解析示意圖	41
圖 25：行動學習教材建置架構圖	42
圖 26：XML PROFILE GENERATOR架構圖	43
圖 27：建立XML文件示意圖之一	44
圖 28：建立XML文件示意圖之二	44
圖 29：VIEW ADAPTOR架構圖	45
圖 30：行動學習教材資源之連結關係	46
圖 31：MVC PATTERN互動關係示意圖	47
圖 32：CONTEXT-SENSITIVE MIDDLEWARE之操作介面	48
圖 33：行動學習教材建置之一	48
圖 34：行動學習教材建置之二	49
圖 35：行動學習教材建置之三	49
圖 36：行動學習教材建置之四	49

圖 37：教材資源之XML文件範例	50
圖 38：教材資源之XSL文件範例	50
圖 39：行動學習教材之HTML範例檔（INDEX.HTML）	51
圖 40：行動學習教材之HTML範例檔（TEXT.HTML）	51
圖 41：MOBiSCO於MIO 8390 之呈現	52
圖 42：MOBiSCO於HP IPAQ H6300 之呈現.....	52

表目錄

表 1：E-LEARNING 國際標準之優缺點分析	14
表 2：欄位判斷之變數	32
表 3：行動設備內文	45

第 1 章 緒論

1.1 前言

IT 技術將如何演變，用以下這句話，便能夠發人深省。

The most precious resource in a computer system is no longer its processor, memory, disk, or network, but rather human attention. [1]

在未來的 IT 環境，電腦系統最珍貴的資源不再是 CPU 的快慢、儲存體的大小或網路頻寬的傳輸效率，而是『人類的注意力』。注意力集中，免於因電腦系統的複雜，導致人類的不專心，造成不必要的危險。同時，也因為注意力集中，才有機會發揮更大的創造力與想像力，激發出無窮的潛力，締造出更燦爛、絢麗與便利的美好明天。為了朝著這個理想邁進，當前的產業技術的兩大趨勢分別為普及運算（Pervasive Computing / Ubiquitous Computing）以及無線網路服務（Wireless Service）。普及運算 [2][3] 一詞可以說是一般性電腦運算的拓展，尚未落實到每個人的思想觀念，其理想是希望能夠達到任何時間、任何地點、任何設備，皆能夠輕鬆取得資訊，所以在未來，我們將不用再關心電腦如何運作，而只需考慮與什麼人做什麼事。故普及運算的發展，勢必將成為我們生活的一部份，並且改變人類的文明生活。

那麼在這樣日漸成熟的 IT 環境，人類的學習模式又會帶來什麼樣的衝擊呢？無線通訊技術為未來教室的學習形式帶來下列條件：學習需求的迫切性、知識取得的主動性、學習場域的機動性、學習過程的互動性、教學活動的情境化及教學內容的整合性 [4]。因此，我們將數位學習（e-Learning）廣泛定義為運用網路促成的學習，包含學習內容的製作、傳遞、擷取、學習經驗的管理、學習社群間

的交流等，與傳統學習形式最大的不同之處在於 e-Learning 將所有與學習有關的活動，透過網際網路、Web-Service、Database 等資訊技術將之整合在一起。

本研究希望能夠搭上目前數位學習的順風車，將現今架構在有線環境的學習平台，加以延伸至無線環境下，以真正達到行動學習（Mobile Learning）！

1.2 研究動機

綜合以上的觀點，本研究歸納出以下三點研究動機作為進一步探討的重點：

1. 在無線環境下 e-Learning 會有怎樣的改變？如何建置一個全方位的行動學習環境？
2. 何謂內文（context）？在行動學習環境下，需考慮到哪些內文？而內文感知（context-sensitive）的特性又為何？如何發展內文感知行動學習環境？
3. 學習教材區分為動態（video、audio）與靜態（word、pdf、powerpoint）兩類。在有限資源的行動環境下，如何提供學習者最佳化的教材呈現？此觀點也是本研究的重心。

1.3 研究目的

本研究希望運用現行的 IT 技術，建立一 SCORM-based 之內文感知行動學習環境（Context-Sensitive Mobile Learning Environment）。一個應用軟體系統的內文（context），就是軟體本身、終端設備、軟體使用者、終端設備環境，可以被偵測的相關屬性。內文感知（context-sensitive）則為系統的一種功能，能夠偵測和分析來自不同的內文，它讓應用軟體可以在不同的環境作不同的動作，亦即針對不同的設備格式，需有不同的互動模式 [5]。行動學習則泛指學習內容可在任何時間、任何地點以及任何種類的設備上作呈現。

故本研究的最終目的有兩點：第一，藉由我們所提出的內文感知中介軟體（Context-Sensitive Middleware），轉換同一份教材於不同的行動裝置（PDA、Smartphone、NB），並且針對呈現方式作最佳化的處理，以期能夠提升現有教材的可再用性（Reusability）。第二，目前的行動教材大部分是以靜態教材為主，本研究則希望能夠將動態教材（video、audio）與靜態教材作結合，藉由教材的多樣性（Diversity），提高學習者的興趣，並且提升學習的品質。

1.4 章節安排

本論文共計分為五章，各章分配架構如下：

- 第一章 介紹本研究之緣起，研究動機、研究目的，作為以下章節的基礎。
- 第二章 整理兩年來所蒐集的研究資料及相關的背景知識，以充分了解目前的研究發展。
- 第三章 根據本研究的系統架構與研究方法，建立我們所提出的『內文感知行動學習環境』，以有效解決當前所遭遇的瓶頸。
- 第四章 本章為這兩年來的研究成果。
- 第五章 針對前幾章所描述的內容作一個總結並闡明未來的研究方向。

第 2 章 相關研究

2.1 e-Learning 背景知識

2.1.1 e-Learning 國際標準

近幾年比較有名的e-Learning國際標準主要包含以下幾種：IMS [6]、IEEE LTSC [7]、AICC CMI [8]、SCORM [9][10][11][12]、LMML [13][14][15]、ARIADNE [13][16] 與ULF [16]，而其中以SCORM最為熱門。本研究乃是遵循SCORM 1.3.1 之規範來建置內文感知行動學習環境。這些國際標準各有其優缺點，整理如表 1所示。以下我們分別詳述之：

➤ IMS

IMS (Instructional Management System) 是一個由產官學界所組成的一個全球性組織。目的是為了在不同的平台之間可以互相使用教材，追蹤學習者的進度，回報使用者的成效，交換學生的紀錄而發展出來的一種開放性規格。IMS為LOM的創始者，有鑑於LOM在實行上的一些困難，IMS仍繼續對LOM進行修改，並交由IEEE LTSC 審定。IMS主要針對IEEE組織下LTSC委員會所訂定學習元件的Metadata，提出表達教材組合模型使用的對應與建議，可以形成基本教材庫，提供教材發展者搜尋取用已存在的適當教材。在IMS中的Metadata包含九種基本元件，包括一般資源描述 (General)、歷史資料 (Lifecycle)、Metadata描述 (Meta-metadata)、技術性資訊描述 (Technical)、教育特性 (Educational)、狀態描述 (Rights)、關聯性描述 (Relation)、使用資源之建議與建構者資訊 (Annotation) 及所屬領域 (Classification)。另外IMS也定義了整個教材的封包格式 (Content Packaging)，可表示一份完整的教材、一門課程，或是一系列課程。

而其中包含兩個部份：檔案實體（Physical Files）以及描述這些檔案如何組合成課程的邏輯組織（imsmanifest.xml）。將教材以封包形式存在的好處在於教材的組成上有最大的彈性，可以將兩份教材拆解、融合、穿插與形成巢狀式的教材結構。

➤ IEEE LTSC

IEEE LTSC致力於推動教學軟體工具之開發與內容維護的標準。LTSC有多個工作小組，其中LOM（Learning Object Metadata）工作小組負責內容標準的制定，其標準藍圖是以IMS為基礎加以修飾而成。

LOM的主要目標是希望藉由此標準的完善，可以讓學習者、教學者和系統軟體更能有效快速的進行搜尋、評估以及資源的獲得與使用。更甚者，可以促進學習資源的交換與分享。LOM定義了一組專門用來描述學習資源的元素，其Metadata共分成九類，包括一般資源描述（General）、歷史資料（Lifecycle）、Meta-metadata、技術性資訊描述（Technical）、教育特性（Educational）、版權描述（Rights）、關聯性描述（Relation）、使用資源之建議與建構者資訊（Annotation）及所屬領域（Classification）。其資料格式共有CharacterString、LangString、DateTime、Duration與Vocabulary五大型態。

➤ AICC CMI

在AICC（Aviation Industry Computer-based Training Committee）組織的CMI（Computer Management Instruction）指引中，針對教材的定義規範了一個課程的課（Lesson）、章（Chapter）包含教學指令集合為20至50分鐘學生學習課程。其資料模型中定義學習物件（Learning Object）為一個基本學習訓練的單位或教學指令群集，其具有獨立性且擁有結構化教材，目標在網頁客戶端操作並提供學

習系統追蹤。

CMI還提供一組資料元件以定義學習資源狀態，提供學習物件與學習管理系統（Learning Management System，LMS）溝通機制。LMS必須在每次學生連上系統時，維護所需資料元件的狀態，而學習物件也是透過這些事先定義的資料元件與不同的教學系統溝通。

➤ SCORM

美國國防部的分散式學習主導計畫（Advanced Distributed Learning Initiative），主要為推動電子學習元件國際標準—SCORM(Sharable Content Object Reference Model)，目的在於提供可再用與分享的課程元件撰寫準則。SCORM將e-Learning系統上的教材視為元件，透過API（Application Program Interface）來操作與分享這些元件，目的是讓網路上不同的學習管理系統透過分享所開發設計的學習教材以達到重複使用的目的。

SCORM基本上遵循IMS教材標準定義，依據IMS訂定之Metadata，可將網路教材打包成可分享之教材元件。SCORM最重要的概念在於可分享教材元件（Shareable Content Object，SCO），其中包含Asset、SCA、SCO與Content Packaging。最左邊每個課程必須有三個檔案，其中Course.HTML為主要檔案，用以描述此課程之一般性Metadata，IMS manifest.XML則描述此課程結構的Metadata，一個課程可以有許多的SCO教材元件。中間每個教材元件包括SCO的主HTML檔與Metadata，以及描述SCO結構的Metadata，一個SCO可由一個以上的實體教學教材檔案組成，如視訊檔、音訊檔等，每個實體教學教材檔案都有自己的Metadata描述，如圖 1所示。



圖 1：教材描述概念圖

SCORM的發展用途主要在於線上教學與遠端學習，以框架（Framework）技術製作出可重複使用的教材元件〔17〕〔18〕〔19〕〔20〕。為了讓學習內容及其系統可符合高階需求之模型，故定義了一套相關技術的規格說明。在教材元件方面描述了CAM（Content Aggregation Model）及RTE（Run-Time Environment），在SCORM 2004 又加入了SN（Sequencing and Navigation）〔9〕〔10〕〔11〕〔12〕。另外，多媒體學習教材也廣泛的應用在數位學習上，故學習元件已不再侷限於文字，因為多媒體教材在訊息的傳遞上，往往能夠比文字更有表達力，目前的研究已提出自動產生SCORM-Based多媒體學習教材的方法〔20〕。雖然SCORM定義了一整套的系統架構，讓學習者可延伸其系統功能，且可與其他系統進行資訊交換與教材共享，然而，SCORM卻無法完整的處理內文感知（Context-Sensitivity）的問題。為了達到智慧型的線上學習，內文感知所扮演的角色日益重要。就現階段而言，有相關的研究提出可在行動裝置上做內文感知的Pocket SCORM〔21〕，但應用上還不是很普遍。

➤ LMML

LMML是德國Passau大學制定的標準，主要是為了配合他們的知識管理系統之網路教材管理所開發的標準，適用範圍是大學教育或職業之訓練教材。LMML為一種XML-based語言，具有相當大的彈性，能延伸到不同應用領域。LMML主要著重於教材內之描述，對於整個平台與系並沒多做任何的規定，LMML亦具有以下特性：

- Model-base之XML架構：能夠適用於不同的領域應用，也能延伸到

XML的其他語言上，例如：MathML, SMIL等。

- 模組分割化：一些分割出來的模組能達到複重使用性。
- 多媒體教材的高度適用性：文件導向式設計，可在原先架構上不斷擴展到其他領域上，並能簡易地將現存文件轉移到不同的格式，也適用於標準的XML編寫工具。

LMML整個架構共分為四部分：

- Structure Modules：主要用來將Content Modules結構化。
- Content Modules：為一Self-Contained元件，內容主要包含了實際的Media Object。
- Structure Objects：其中包括了Text、Image、Sound與Animation等物件元素。
- Media Objects：結構化的物件，例如ulist、olist、table等元素。

➤ **ARIADNE**

ARIADNE (Alliance of Remote Instructional Authoring and Distributed Networks for Europe) 起源於西元 1996 年，由歐洲數個大學所組成之聯盟，旨在藉由e-Learning的方式幫助專業人員進修，這套標準也考量了歐洲的文化歷史背景，所以強調多國語言與多國文化的包容性。其規格是參考LOM而來，但整體而言卻較LOM稍微精簡，且其支援IEEE LTSC與IMS標準。ARIADNE的Metadata共分為 6 大類：一般資源描述 (General)、資源的語譯描述 (Semantics of the Resource)、教學文件屬性 (Pedagogical Attributes)、技術特性 (Technical Characteristics)、使用條件 (Condition for Use) 與再詮釋資料 (Meta-metadata)。

ARIADNE提供了一個完整的e-Learning平台以及許多教材編輯工具來幫助教師製作教學文件，而在教材的儲存與交換上，ARIADEN建構了一個大型的分散式系統（Knowledge Pool System）作為教材資料庫，並藉由ARIADNE的工具來達到教材管理和資料交換與再使用（Reuse）之目的。在教材編輯上，ARIADNE把教學文件分為兩種：Active Document與Expositive Documents。Active Document可讓使用者與電腦產生互動，如Simulator（模擬、實驗）、Auto-evaluation（自我評估練習）等。而Expositive Document則是比較單純地接收知識，例如：文字、影像等。將編輯好的教材產生一份標頭檔資料，接著存入Knowledge pool System中。

➤ ULF

ULF（Universal Learning Format）是以XML和RDF為基準的一套教材定義格式，由SABA公司根據一系列學習系統所統整而定義出來的，主要是用以處理線上學習各種資料的整合性標準，其目的為提供一套完整的規格描述，並與目前其他標準作結合。ULF包括下列五個模組，如圖 2所示。

- Learning Content Format：描述教材目錄、內容與試題的標籤定義。
- Catalog Format：描述教材關係、一般性資訊與資源搜尋的標籤定義。
- Competency Format：描述個人學習成果的標籤定義。
- Certification Format：描述個人學習成果的標籤定義。
- Profile Format：描述個人基本資訊與學習歷程的標籤定義。

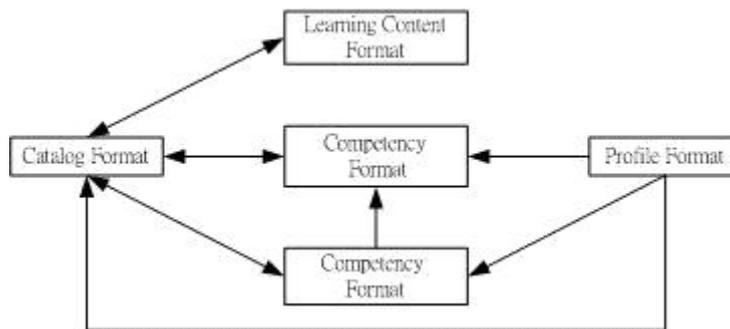


圖 2：ULF模組關係圖

表 1：e-Learning 國際標準之優缺點分析

國際標準	優點	缺點
IMS	<ul style="list-style-type: none"> ● 教材標準之制定與規劃相當完善，例如 Simple Sequencing 與 Content Packaging ● 制定的 Metadata 被廣泛採用，因此可與許多教材標準相容，例如 SCORM 	<ul style="list-style-type: none"> ● 偏向概念次的提出，實際系統細節的實作尚未非常完整
IEEE LTSC	<ul style="list-style-type: none"> ● 具有多國語言的特性，有助編碼與解碼的進行 ● 廣泛地被採用，例如 IMS、ARIADNE、ADL SCORM 等皆是採用此標準 	<ul style="list-style-type: none"> ● 某些分類元素之宣告仍不夠精細，無法做更深入的延伸與分析，例如教育特性之元素定義即無法有達到本土性教材之支援
AICC CMI	<ul style="list-style-type: none"> ● 高度操作性：不同 CMI 系統間的課交換格式定義完整，只要符合 AICC 規格，就可達到教互相交換整合的目的 ● 簡易重組教材：一份課程由許多物件組成，可根據紀錄檔找到需的物件，故教材可輕易重組再使用 ● 利用文字檔簡易交換資料：CMI 與 CBT 間的通訊只需一個 ASCII 文字檔即可，該檔包含與學生相關的資訊，如學習路徑、學習成果等，故管理較簡單 	<ul style="list-style-type: none"> ● AICC 規格幾乎都是在定義 CMI 與 CBT 及其他 CMI 間課程交換的格式，對於課程的設計與教學文件的格式太多著墨，可以說是沒有 Metadata 與 Content Packaging 之定義，對於 Course Structure 規格也過於簡單
SCORM	<ul style="list-style-type: none"> ● 統合現有各大國際標準，例如 IMS 的 Metadata 與 Content Packaging、IEEE Metadata Dictionary、AICC 的 Content Structure、Data Model 和 	<ul style="list-style-type: none"> ● 無測試的標準：SCORM 把試題當成一種 Asset，對於題型、配分等皆無詳細定義，故難以為學習者評量 ● 不同教材間無法分析：SCORM 並

	<p>Communications</p> <ul style="list-style-type: none"> ● SCORM 建立一個可供測試的平台，供使用者批評、測試 ● SCORM 定義了一套完整系統架構，讓使用者可延伸系統功能，且可與其他 SCORM 系統進行溝通與教材交換 	<p>無針對教材分析做定義，故在學習效果的評估中，無法分析教材間的難易度是否契合，或學習路徑的安排是否適當</p> <ul style="list-style-type: none"> ● 使用者無法動態的選擇教材：SCORM 教材的安排由系統指定，而系統卻是由教材編輯者所決定
LMML	<ul style="list-style-type: none"> ● 擴充容易：以模組化方式設計，只要在各模組新增所需之 Elements 即可完成擴充 ● 結構內容穩固：教材編輯者可使用 Structure Module 來對不同的使用者規劃不同之學習內容 ● 內容彈性大：每個 Content Module 使用的元件可以不同，所以使用的物件也具備選擇性 	<ul style="list-style-type: none"> ● 低階的媒體物件加入不易：低階物件加入時，所有上層的內容模組皆需考慮是否需要使用此物件 ● 使用的便利性不足：因為階層性不夠明顯，所以編輯者容易造成混淆 ● 著重於教材內容之標記，卻沒有對系統任何的規劃，所以比較類似傳統教材的標記工作
ARIADNE	<ul style="list-style-type: none"> ● 良好之教學標準架構與豐富的編輯工具：提供豐富的教材編輯工具給授課者來快速地編輯教材，且可利用課程編輯軟體設計出符合需要的教學流程 	<ul style="list-style-type: none"> ● 未明確定義教材結構，Metadata 之定義不夠詳細完整
ULF	<ul style="list-style-type: none"> ● 與 IMS、ADL、LEEE LTSC 等標準能某種程度的互通 ● 將教學環境及資源規格劃分為五個部份且每一部份各有其專屬規格，以達到模組化效果 ● 教材中的題型、評分設計詳盡，為其它標準所缺 ● 具有認證機制（Certification Format）、分級機制（Competency Format）與編目機制（Catalog Format）可提供學習者認證並且設定不同教學等級等功能 	<ul style="list-style-type: none"> ● Learning Content Format 偏向單純將傳統教材作標記，缺乏教材包裝、巢狀組織、互通、交換方面的規格說明 ● Structure Element 及 Content Element 與其它標準比較起來，較顯薄弱 ● 學科未作特殊化，故其 Metadata 為一般性的資訊 ● 與其他標準之包裝理念不同，因此教材轉換困難

2.1.2 學習內容管理系統

一個學習管理系統 (Learning Management System) 的目的，主要是為了能夠簡易的管理組織內部的學習方案 [22]。對於學習者而言，學習管理系統有助於規範與計劃他們的學習過程，並且讓他們在學習中可以相互的聯繫與合作。針對管理者而言，學習管理系統可以幫助他們鎖定、追蹤與分析學習者在組織內部的學習狀況。然而大多數的學習管理系統並無法讓管理者來編輯教材，這也就是為什麼大部分的學習管理系統都需要額外的編輯工具，或者透過其他的合作廠商來產生所需的教材。個別的教材編輯工具譬如Macromedia's Dreamweaver就可以用來建立客制化的教學內容。我們可以清楚地了解在學習管理系統裡的教材與其內部的教學單元，均屬於不同課程中的部份，如圖 3所示。因此，我們只能就『課程』的層級來重複使用這些資源。

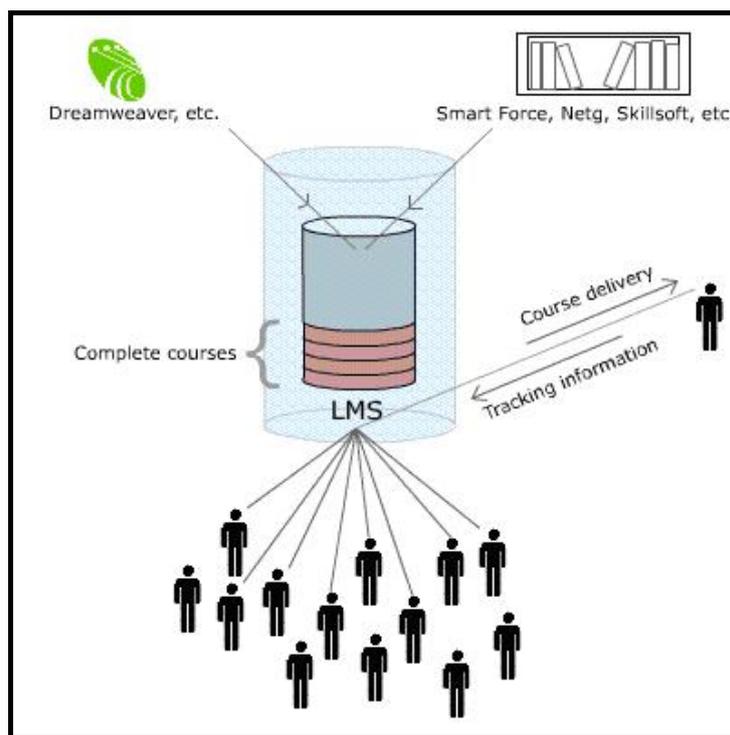


圖 3：LMS 示意圖

內容管理系統 (Content Management System) 一詞主要被用於線上刊登教材，其目的是希望能夠簡單的建立與管理線上的教學資源，包括文章、報告書、影像圖片、廣告、大標題等 [22]。例如我們有一個新聞站台，它有來自 100 個不同

國家的記者，必須不定期的在網站上刊登文章。為了避免大量重複性質的資料被重複刊登，在內容管理系統的管理上必須符合以下兩個要素：

- **把內容從視覺呈現上分割出來：**記者需要集中把他們的內容譬如 MS-Word 文件傳送到一個定點而不需要考慮文章中的圖片出現位置、多大的標題列等問題。所有同類型的內容呈現，排版上都會有刊登的模板（Publishing Templates）來處理。例如第 25 位記者與第 88 位記者所寫的文章可能具有相當大的重複性，相同的資源卻可以展現不同的排版分別符合兩位記者的需求。
- **執行工作流程：**文章在記者們送出後，在刊登前一定會先經過編輯者的處理。

在內容管理系統中，完整的文章是收集許多獨立的內容元件所組成的。這些內容元件被用於不同的新聞可能會有不同的呈現。如圖 4 所示，我們可以清楚地了解這些內容元件式各自獨立的單元，因此就教學內容的角度而言，重複使用性可以由課程的層級提升到內容元件的層級。這些內容元件在數位學習領域，被稱為學習物件（Learning Object）或者可再用學習物件（Reusable Learning Object）。

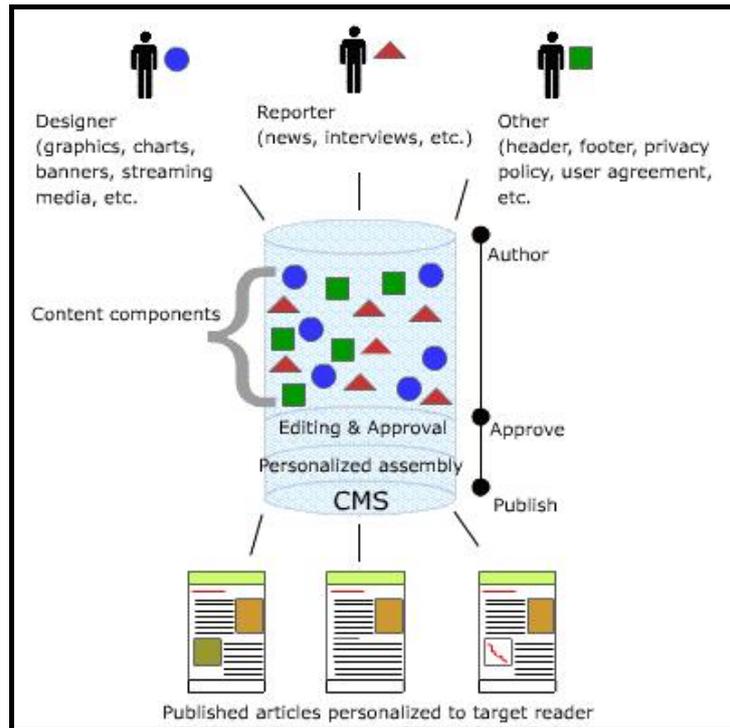


圖 4：CMS 示意圖

對於可再用學習物件的定義，各方學者的認定不同。有些認為一個學習物件可以是一張影像（Image）或者是一份視訊檔案，有些則認為學習物件應該是符合某特殊目標的教學資源的集合。因為學習物件是教材最小單元，因此可以被混合或是以其他方式來組成更大的個人化教材，例如課程、章節等。如圖 5 所示，學習內容管系統（Learning Content Management System）〔22〕〔23〕是指可以編輯、核准、發布或刊登以及管理學習內容（即學習物件），也就是結合了學習管理系統以及內容管理系統的功能。

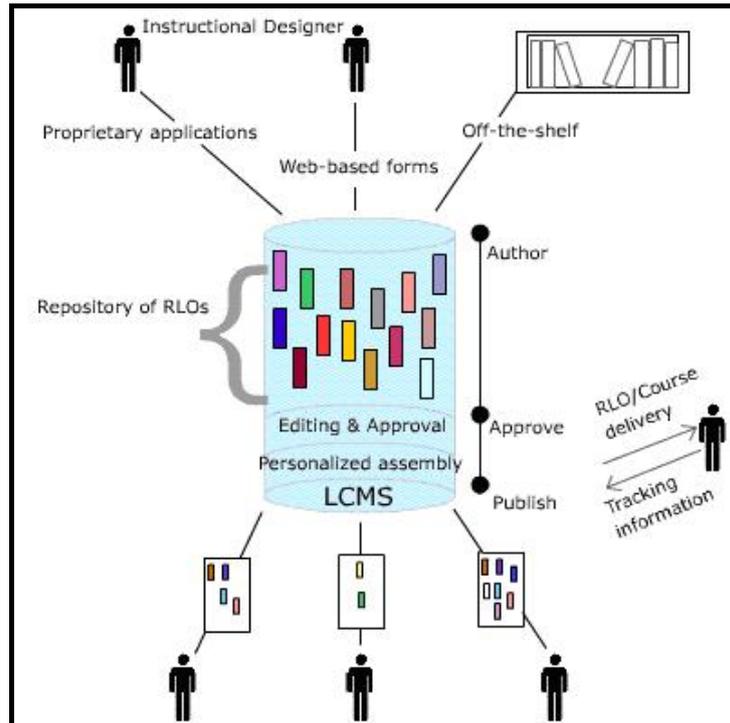


圖 5：LCMS 示意圖

2.2 Context-Sensitive 機制

Context 從字面上的解釋來看有作為文章上下文脈絡之意，從字典上的解釋則是稱某些事物發生或存在的關連性因素（Interrelated Conditions），然而單就通用性的解釋並不適合描述我們想要發展的 Context-Sensitive 系統，有鑑於此必須詳加檢視以往針對 Context-Sensitive 機制所稱的 Context 因子究竟所指為何並進行深入性的探索。

2.2.1 Context 之定義

根據 Dey and Abowd (1999) 的研究認為 [24]，人與人的溝通牽涉到許多複雜的背景與環境因素，即所謂的 Context 因子，例如一場研討會中的與會眾人，縱使台上的主講人口沫橫飛許多專業術語理應不會阻礙他們對所談內容的理解，原因在於會參與這種研討會的人彼此間都對這些主題有一定的瞭解，能夠快

速理解並分享共通的語言，例如專業術語，自然溝通上並不會有太大障礙，然而這些我們認為理所當然的溝通方式，對於要對應到任何運算設備而言卻非常困難，因為依賴某種主題或情境的資訊運算設備都不能自動獲得以及理解要如何使用。在過去的歷史中，忽略這些 Context 因子系統設計者未必會聯想到是應用系統未盡完善的主因之一，原因在於設計者總是以為依背景資訊調整系統組態是使用者的責任，因此應用程式不需要主動地認知到使用者周遭的環境資訊，例如使用者目前的位置在美國或是台灣，這種地理性差異的背景因素，需要系統轉換不同的語文輸入方式或是不同的度量單位，然而系統設計時主觀認為這些設定是使用者應該自行調整的任務。這種設計理念直到分散式運算、行動運算或可攜式運算方式的逐漸普及之後開始遭遇到困難，例如使用者所處環境的 Context：地點、時間、活動、使用設備等，對於行動化的運算設備而言隨時都在變動，例如進出劇院、電影院或會議室，如果仍要使用者不斷地切換組態設定來配合當下的環境似乎太不尊重人性，而且這十分影響使用者使用行動運算設備的意願。

然而到底 Context 是什麼呢？過往的研究有從列舉（enumeration）及同義詞（synonym）兩方面分別來加以解釋。主張用列舉方式的 Schilit and Theimer（1994）認為地點、周遭的人及物件、或他們的改變情形為 Context [25]；Brown（1997）等人則是認為地點、周遭的人及物件、時間、季節、氣溫 [9]；Ryan（1997）等人則以地點、環境、可識別的實體（Identity）、時間為主 [26]。利用同義詞的另一派，Franklin and Flashchbart（1998）認為是使用者的情境（situation）[27]；Ward（1998）等人傾向支持是應用程式環境的狀態（state）[28]；Rodden 等人（1998）則以應用程式的組態設定（setting）作為定義 [29]。

從上述的討論中可知，無論採用列舉或是同義詞解釋的方式似乎都無法兼容各家說法，因此 Dey and Abowd 曾參考上述各家定義，認為 Context 是為『與使用者及應用程式之間互動有關的資訊，任何足以描述人、事、時、地、物等實

體 (entity) 的資訊皆屬之」[30]，正如前述所言 Context 是用來簡化使用者及應用程式間互動之用，因此與其有關的相關資訊自然都需納入討論，不過該納入考慮的 Context 仍應視系統特性而定。再者，由列舉方式定義的大部分研究其實都大同小異，其中又以地點 (location) 一項重複次數最多，這也足見 Context-Sensitive 機制其實是發軔於對地理環境的思考，至於其他足以影響人機互動的大抵是對環境或使用者狀態為主的資訊。Dey 等人將地點 (location)、識別(identity)、活動(activity)、時間(time)列為主要的 (primary) context 項目，這些主要項目會帶出其他的次要項目，例如個人的識別會帶出通訊方式、個人偏好、好友名單等，如此將 Context 簡化在此四大項進行思考。再進一步，Chen and Kotz (2000) 參考 Dey and Abowd 的定義，認為為了要強調行動運算 (Mobile Computing) 環境的特色，將 Context 定義又分為兩部分，一是為了決定行動應用 (Mobile Application) 的行為 (behavior) 而需要被強調的周遭環境特徵 (characteristic)；一是與應用相關但非用來決定其行為的部分。兩者共同包含有環境狀態 (environment states) 及組態設定 (settings) 的資訊 [31]。

2.2.2 Context-Sensitive 之定義

了解何謂 Context 後，那麼系統要如何運用 Context 資訊來簡化與使用者的互動呢？或是要如何讓系統從眾多 Context 資訊中察覺最關鍵的項目，進而提供關鍵的個人化服務？首先設計人評估某些需要被特別觀察的 Context 項目，並定義取得該 Context 項目的方式，於是系統就會不間斷偵測此 Context 因子的變化情形，在偵測到 Context 狀態有變化時系統會依其變化自動改變自身組態以提供相應的服務，也就是依照 Context 改變而提供最適化 (adaptive) 的服務內容，具有這種功能的系統一般認為具有 Context-Sensitive 的能力 [32][33]。然而，系統改變要到何種程度才算 Context-Sensitive，相關論述有二：一是僅針對 Context 加以回應 (respond) 的 Using Context；以及除了回應之外還要能依 Context 不同

提供不同適合服務的 Adapting to Context。前者有 Hull (1997) 及 Pascoe (1998) 等人，認為 Context-Sensitive 是計算設備有能力偵測、感知、解釋、回應使用者周遭環境及設備本身的觀點；Dey (1998) 將 Context-Sensitive 限制在人機介面互動上；Salber 等人 (1998) 認為 Context-Sensitive 就是有能力根據即時偵測到的 Context 提供最具彈性的計算服務 [33]。

Adapting to Context 的擁護者定義 Context-Sensitive 要能基於系統及使用者的 Context 而動態的改變或最適化程式本身的行為 (behavior)，限制 Context-Sensitive 必須是系統能夠改變自身行為方可。其實系統若能自動化回應 Context 的觀點亦不妨稱之為 Context-Sensitive，重點應放在是否能夠自動感知、回應上，至於提供最適化服務自然是最好，但不應作為 Context-Sensitive 的必要條件，因此 Dey and Abowd (1999) 定義屬於 Context-Sensitive 的系統，是因為利用 Context 來提供資訊或服務，而這些資訊與服務與使用者目前任務相關，而未必需要涉及改變系統行為的階段 [33]。

除了定義外，Context-Sensitive 究竟有哪些類型，Schilit 等人 (1994) 認為 Context-Sensitive 依照系統反應 Context 改變的自動化程度高低。Pascoe (1998) 提出類似 Schilit 的 Context-Sensitive 類型，不過增加了使用者可以針對 Context 內容附加數位資訊的能力。Dey 等人 (1999) 整合上述兩者的說法，認為 Context-Sensitive 主要分成三項，主要負責資訊或服務的展現，執行自動化服務和賦予使用者增加 Context 資訊的能力 [33]。

2.3 XML & XSL

XML 是由 W3C (World Wide Web Consortium) 所制定而成的標準語言 [34]。XML Schema 則定義了 XML 文件內的結構與限制，以用來確認 XML 文件是否可驗證的方法之一，符合 XML Schema 的 XML 文件才是可驗證的，否則就是無

效的。XSL (Extensible Stylesheet Language) [34] 是一種用來描述樣式版 (Stylesheets) 的語言，由以下三個部分所組成：

- XSLT (XSL Transformations) : 使用 XSL 格式指令將 XML 文件轉換成其他以文字為主的文件，其轉換形式可以是 XML、HTML、純文字或是以文字為主的文件。
- XPath (XML Path Language) : XSLT 用來存取 (access) 或參考 (refer) XML 文件的其他部分。XPath 也可被 XML 用來當成連結描述。
- XSL Formatting Objects : 這是一種 XML 的字彙，可被用來說明結構化的語意。

2.4 Design Pattern MVC

在發展已成熟的工程領域中，通常都有一些用來描述對已知問題之解決方法的手冊。舉例來說，汽車設計師在設計汽車時不會使用物理定律而是他們會重用過去已經成功的標準設計。為了在效能上增加少許的改善，而從無到有整個重新設計，所得到的效益將遠不及所花費的成本。而在軟體工程界，針對一些常見的問題，也有許多成功的解決方法 (Solution)，設計樣式 (Design Pattern) 企圖描述這些解決方法，讓工程師在遇到類似的問題時，能夠很快地將這些成功的方法應用在他們自己的問題上。通常一個 Solution 能夠被視為 Design Pattern，表示這個 Solution 是經過實證的，而非只是一個新穎的概念而已。

MVC (Model-View-Controller) [35] 是一種相當著名，且廣泛被採用的設計樣式，談起它的歷史，最早可追溯到 Smalltalk (一種物件導向程式語言，由 Alan Kay 所發明) 語言的 Smalltalk-80 版本。在 Smalltalk-80 中，使用者介面的主要概念就是 MVC。MVC 的基本概念就是將使用者輸入、資料本身以及資料的呈現三者分離，各司其職，如圖 6 所示。以下分別描述之：

- **Model.** 管理應用領域 (Application Domain) 的資料、執行資料的處理或轉換、回應請求資料的要求 (通常來自 View) 以及回應變更資料的要求 (通常來自 Controller)。Model 對 View 與 Controller 一無所知，它們透過特定的介面 (Interface) 與彼此溝通。
- **View.** 負責顯示 Model 中的資料，也就是將資料做視覺化的呈現。同樣地，View 透過特定的介面與 Model 溝通。
- **Controller.** 提供改變 Model 中資料的機制。Controller 解譯使用者所發出的鍵盤或滑鼠事件，以便告知 Model 資料已改變的事實。Controller 也是透過特定的介面與 Model 溝通。

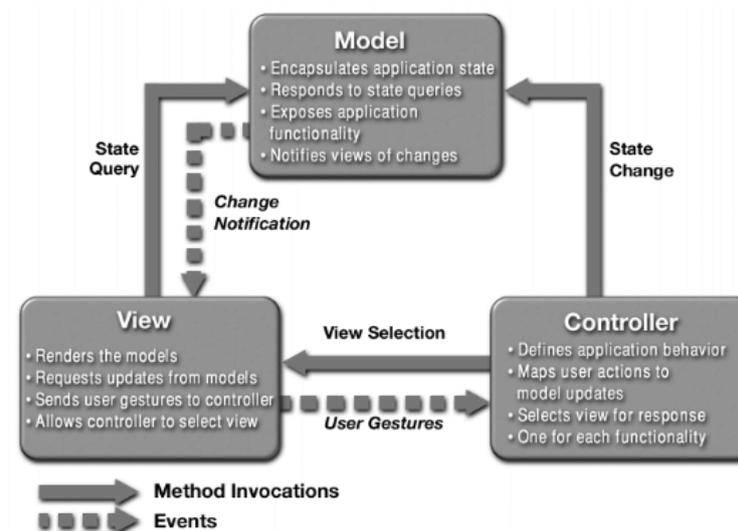


圖 6：MVC Pattern 概念圖

第 3 章 內文感知行動學習環境

行動學習泛指學習內容可在任何時間、任何地點以及任何種類的設備上做呈現。現今，學習內容管理系統 (Learning Content Management Systems) 已發展出便利的教材編輯工具 (Authoring Tool) 與即時的學習環境，以協助老師快速地製作教材以及讓學生有效率地線上學習 (on-line learning)。然而，大部分學習內容管理系統之呈現環境卻還是以桌上型裝置為主，例如桌上型 PC 或 Notebook，而非行動裝置，例如手機或 PDA。透過行動裝置來學習新的知識，在普及運算環境 (Ubiquitous Computing Environment) 日趨成熟的情況下，是可行且必要的。

本論文提出一個內文感知的中介軟體 (Context-Sensitive Middleware)，透過此中介軟體，可將一份原本只能呈現在桌上型裝置的教材，依據不同行動裝置的規格作最適化的處理，以達到最佳化的呈現方式。教材的種類大致上可區分為靜態文件 (word、powerpoint、pdf) 與動態文件 (video、audio)，一份行動學習教材則是靜態文件與動態文件的整合。

第三章是本研究的核心，靜態文件與動態文件的解析過程將陸續探討。一開始我們先介紹內文感知行動學習環境的系統架構。在 3.2 節與 3.3 節，則針對一份靜態文件的解析與存取作深入的描述。在 3.4 節與 3.5 節，則針對行動學習教材的解析與建置加以說明。

3.1 系統架構

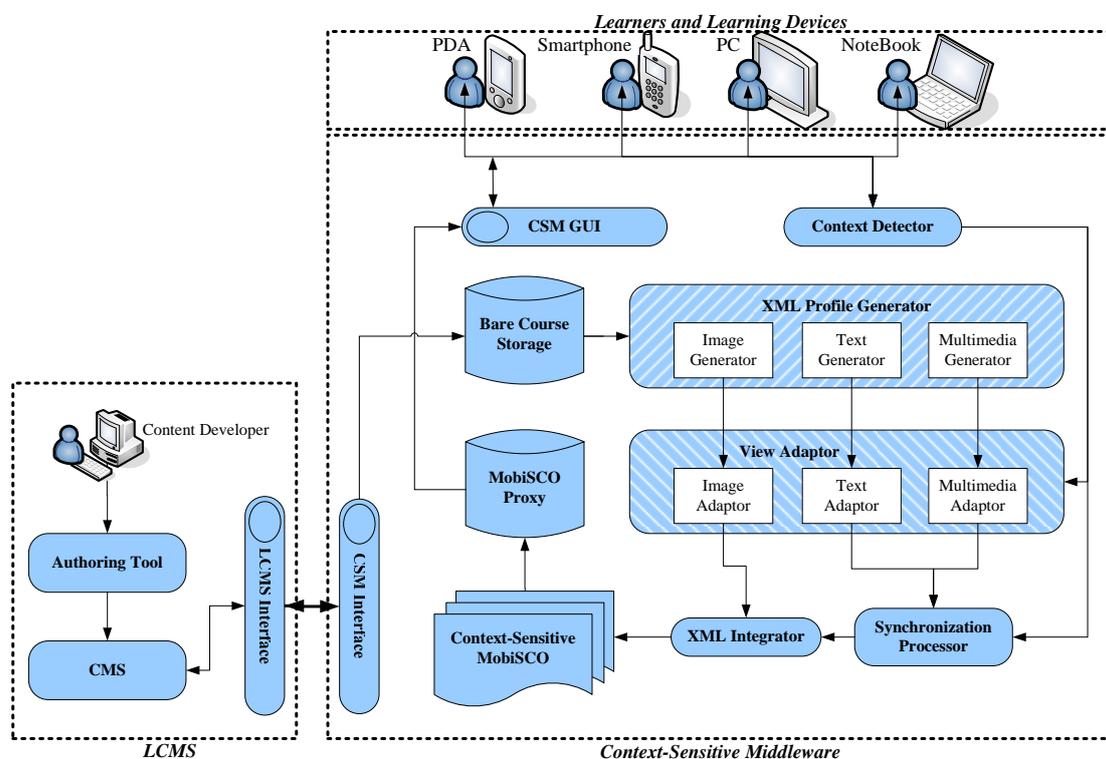


圖 7：內文感知行動學習環境之系統架構圖

在設計上，本論文採用 MVC 模版來規範整個系統架構。為了提升學習教材的可再用性（Reusability）並且與國際標準接軌，我們遵循 SCORM 1.3.1 的國際標準，SCORM 的精神主要是將學習教材分成 Asset、SCO、SCA 以及 Content Packaging，利用 XML 來描述這些元素，並且建立元素之間的連結關係。

內文感知行動學習環境主要分為三個部分，如圖 7 所示：

- LCMS. 學習內容管理系統（Learning Content Management System）的簡稱。此系統主要的作用是教材的製作與管理，教材編輯者（授課老師或編撰委員）利用教材編輯工具（Authoring Tool）將製作好的教材放置在教材管理系統（Content Management System）。此系統在內文感知行動學習環境中，負責提供原始教材（Original SCO），利用 LCMS Interface

作為與 Context-Sensitive Middleware 溝通的介面。

- **Learners and Learning Devices.** 在內文感知行動學習環境的概念中，知識是無所不在且隨手可得。故學習者可透過不同的手持行動裝置（PDA、Smartphone、PC、Notebook）來學習新的知識。根據行動裝置的內文（Context）不同，呈現的方式也會有所差異。
- **Context-Sensitive Middleware.** 在內文感知行動學習環境中，此系統所扮演的角色相當重要，除了接受由 LCMS 所提供的原始教材外，並將同一份原始教材，根據行動裝置的不同加以處理，以轉換成不同的行動學習教材。

在本論文中，內文感知內容呈現（Context-Sensitive Content Representation）為研究的重心。在內文感知行動學習環境之系統架構中，Context-Sensitive Middleware 可將一份原始教材，根據設備內文（Device Context）的不同，轉換成不同的行動學習教材，故此子系統是本研究最核心的部分。為了更了解 Context-Sensitive Middleware 的運作模式，首先針對其內部的模組來介紹，歸納如下：

- **CSM Interface.** 在 Context-Sensitive Middleware 中 CSM Interface 負責與 LCMS 中的 LCMS Interface 作溝通，透過此介面來獲得 LCMS 所提供的 SCORM-based 之原始教材。
- **CSM GUI.** 此模組為 Context-Sensitive Middleware 與學習者溝通的圖形化使用者介面（Imageic User Interface），在我們的內文感知行動學習環境中，呈現的方式是網頁模式（Web-based），故 CSM GUI 是一入口網頁（Portal）。

- **Context Detector.** 顧名思義，此模組負責內文的偵測，本研究則以行動裝置的設備內文為主要偵測的對象。當使用者透過不同的行動裝置連到 CSM GUI 的同時，Context Detector 便開始偵測該設備的內文，依據設備內文的不同，產生出多樣性的呈現方式。
- **Bare Course Storage.** 從 LCMS 中獲得的原始教材，將放置在 Bare Course Storage。原始教材的種類繁雜，其中包括靜態文件（word、powerpoint、pdf）與動態文件（video、audio）。本研究欲將靜態文件與動態文件整合成一份行動學習教材，例如：powerpoint + video。讓我們將教材資源的部分作更明確的劃分，主要可分為三種：影像（Image）、文字（Text）、多媒體（Multimedia）。
- **XML Profile Generator.** 一份行動學習教材需符合 SCORM 標準，故我們透過 XML 來描述以上三種教材資源。XML Profile Generator 內部細分成三個子模組：Image Generator、Text Generator、Multimedia Generator。
- **View Adaptor.** 根據前一個模組所產出三份 XML 文件，加上 Context Detector 偵測到的設備內文，分別對圖形、文字與多媒體作最適化的處理。View Adaptor 內部亦細分成三個子模組：Image Adaptor、Text Adaptor、Multimedia Adaptor。
- **Synchronization Processor.** 靜態文件與動態文件整合成一份行動學習教材是我們的目標。以 powerpoint + video 為例，學生透過手持行動裝置，一方面可瀏覽投影片內容，另一方面還可觀看老師上課的情形。透過 Synchronization Processor 可達成此一目標。
- **XML Integrator.** 簡言之，教材資源的整合與行動學習教材的建置，透

過 XML Integrator 來達成。以技術層面來解釋，XML 文件需搭配 XSL 所提供的樣式 (Stylesheet)，才能轉換成學習者所需要的呈現模式，細節部分，往後章節將深入介紹。

- **Context-Sensitive MobiSCO.** 一份行動學習教材的建置，是由原始教材資源搭配設備內文所整合，故我們稱為 Context-Sensitive MobiSCO。
- **MobiSCO Proxy.** 建置完成的行動學習教材會被存放在 MobiSCO Proxy，以便學習者日後要瀏覽同一份教材，直接便可透過 MobiSCO Proxy 取得，無須重新建置以提升系統的效率。

透過以上各模組的介紹，對於內文感知行動學習環境有了大致的了解。基本上，Context-Sensitive Middleware 就是用來分析不同種類的原始教材資源，並根據設備內文的不同，產生一份客制化的行動學習教材。而在此之前，我們得先了解一份靜態文件是如何解析與存取，以提升現有教材資源的可再用性，於 3.2 節與 3.3 節詳述說明之。

3.2 靜態文件解析

本小節 (3.2) 與下一個小節 (3.3)，主要的重點放在靜態文件部分的解析與存取。一般而言，靜態文件最基本的組成元素為影像 (Image) 與文字 (Text) [38]。簡言之，一份靜態文件的解析過程就是針對這兩種元素來處理，處理後的結果再透過 Content Management System (CMS) 存放在資料庫，此部分包含在整個內文感知行動學習環境的 LCMS，如圖 8 所示。

當教材編輯者 (Content Developer) 透過 Static Doc Interface 將手中的靜態文件 (word、powerpoint、pdf) 上傳至 Authoring Tool，此時 Authoring Tool 中的

Static Doc Agent 則開始解析靜態文件的架構及其組成元素。Static Doc Agent 利用 Image Extractor 與 Text Extract 將這些元素擷取出來，並依據 SCORM Content Aggregation Model[9][10][11][12]內所定義的元素層級，呼叫 XML Controller 建立一份 XML 文件 (Manifest File) 來描述其架構關係。我們將一份靜態文件定義成一份 Content Packaging，文件裡的每一頁則定義成一個 SCA 或 SCO，每一頁中的文字與影像則定義為最基本的 Asset。

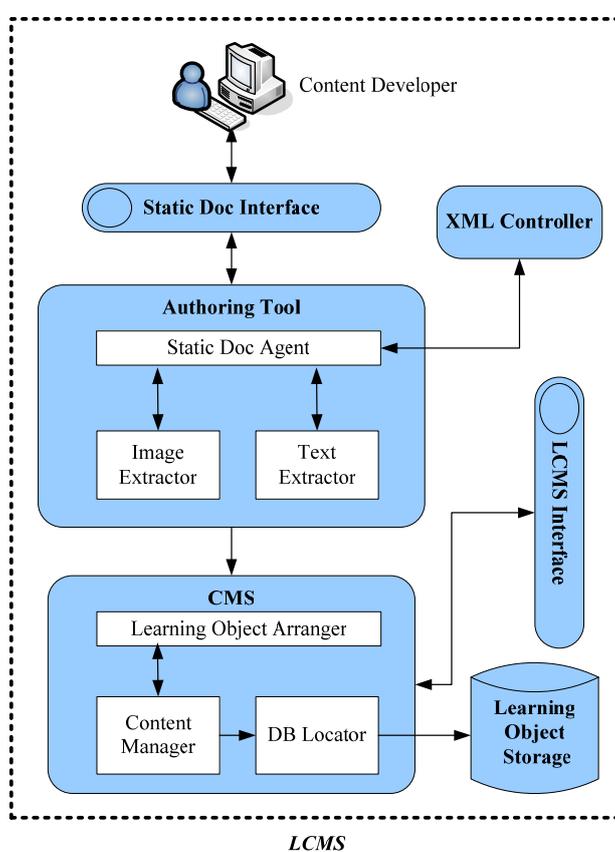


圖 8：靜態文件解析架構圖

當XML文件建置完成，Authoring Tool中的Static Doc Agent便會將工作權交給CMS中的Content Manager來處理。Content Manager主要有兩個工作：首先是呼叫Learning Object Arranger，將這些元素加以分類並且定義彼此之間的關係。例如一份靜態文件中的某段文字 (Text) 是用來解釋某張影像 (Image)，那麼這兩者必然存在著相依關係，故視為一組單元 (Unit)，亦是文件內文 (Document Context)

的一種。所謂的文件內文，是指一份靜態文件中的某些元素間存在著高度的關聯性與相依性，如圖 9 所示。之後再呼叫 DB Locator，將這些元素依序存放到資料庫。

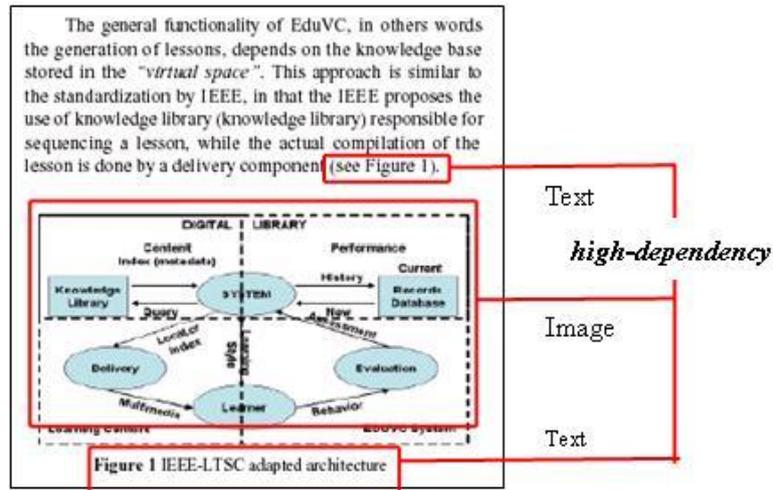


圖 9：元素相依關係示意圖

就技術層面而言，我們透過轉檔程式將一份靜態文件轉換成 HTML，Static Doc Agent 透過 Image Extractor 與 Text Extractor 來解析 HTML 的標籤，細部的處理上，可分為以下三個部分：

- 版面佈置：一份靜態文件的版面佈置可分為三種，分別是『單欄』、『雙欄』與『多欄』，如圖 10 所示。我們利用欄位之間的空白區域來判斷欄數。判斷的方式一開始先定義相關的變數，如表 2 與圖 11 所示，我們得到了 $C = \{C_{end} - C_{start}\}$ ， $S \in C$ ， $C \in [D_{start}, D_{end}]$ ， $Type = \{T_1, T_2, T_3\}$ ，並利用以下的虛擬碼來辨識靜態文件的樣式。



圖 10：靜態文件版面佈置種類

```

begin
  for (Search S from Dstart to Dend)
    if there is a S in C2
      if there is a S > Thre in C3
        Type = T3
      end
    else if
      there is a S > Thre in C1
        Type = T2 ;
    else
      Type = T1
    end
  end
end
end

```

表 2：欄位判斷之變數

名稱	意義
<i>S</i>	空白區段或此區段內大部分為空白 (Space)
<i>C₁</i>	將文件縱向切為兩等份，中心部分的區段
<i>C₂</i>	將文件縱向切為三等份，介於第一等份與第二等份間的區段
<i>C₃</i>	將文件縱向切為三等份，介於第二等份與第三等份間的區段
<i>C_{start}</i>	某區段的起點
<i>C_{end}</i>	某區段的終點
<i>D_{start}</i>	文件的起點
<i>D_{end}</i>	文件的終點
<i>Thre</i>	空白區段的門檻值 (Threshold)
<i>T₁</i>	文件為單欄樣式
<i>T₂</i>	文件為雙欄樣式
<i>T₃</i>	文件為多欄樣式
<i>Type</i>	文件樣式的型態

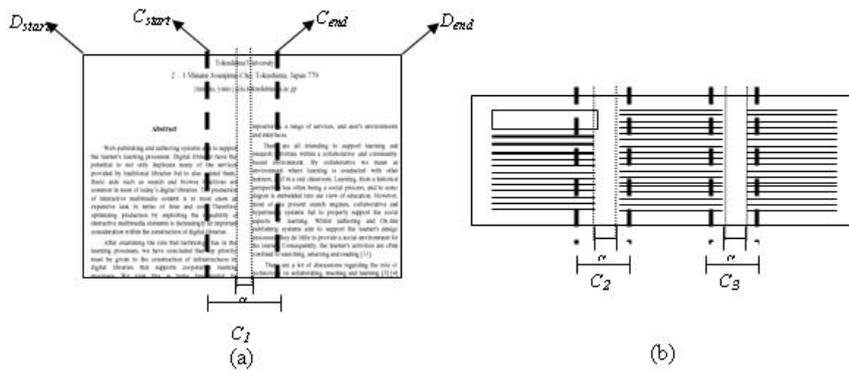


圖 11：(a) 雙欄樣式 (b) 多欄樣式

- 影像解析：Static Doc Agent透過Image Extractor來解析影像，在解析的過程中將擷取出來的影像依序編號，其目的是為了快速重新排版，如圖 12 所示。

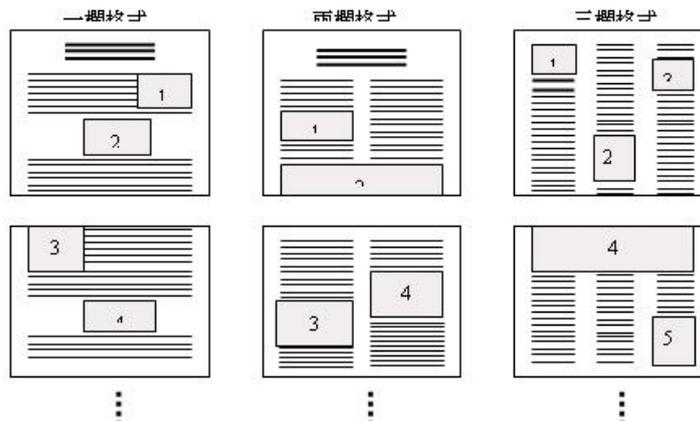


圖 12：影像編號示意圖

- 文字解析：一份靜態文件的文字，可分為『本文部分』與『其他部分』。本文部分為文章的實際內容，比例佔大部分。其他部分則包括標題、註解及其他資訊，故通常會以特殊字型來排版。Static Doc Agent透過Text Extractor來解析這兩部分的文字。就技術層面而言，首先將一份靜態文件轉換成HTML，再根據其中的font標籤加以判斷，例如。而判斷的方法首先是統計此份文件中size出現最多次的數值，並以此作為門檻值，小於或等於此數值的文字元素被視為『本文部

份靜態文件，Context-Sensitive Middleware中的CSM GUI為學習者與系統溝通的橋樑，同時Context Detector偵測其設備內文，以達到最佳化的內容呈現，如圖 14 所示。所謂最佳化的內容呈現，主要涉及以下幾個議題：

- 網頁切割與網頁超連結：此議題主要是解決行動裝置記憶體不足的瓶頸，如圖 15所示。其作法是當原始網頁達到 4/5 的記憶體容量時，就將原始網頁切割成兩個子網頁 P_1 與 P_2 ，並建立 P_1 與 P_2 間的雙向超連結，如圖 16所示。

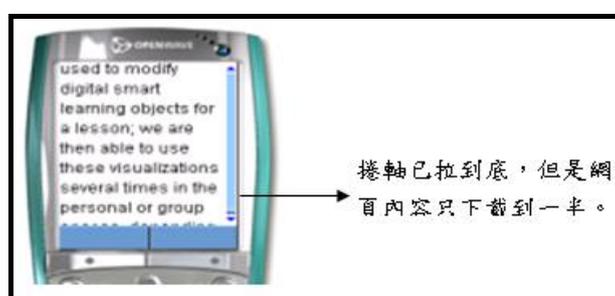


圖 15：記憶體不足示意圖

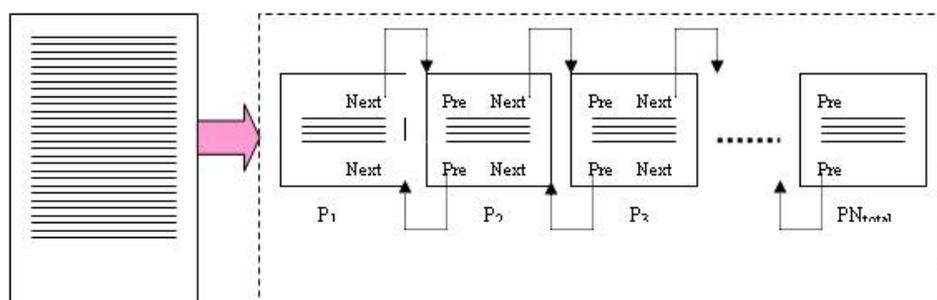


圖 16：網頁切割示意圖

- 影像之最適化：原始影像需考量行動裝置的螢幕大小來作調整，另外則需考量行動裝置本身的解析度是否支援彩色模式。透過 Context Detector 來獲得這方面的資訊，故此議題也是設備內文的一種。
- 雙向捲軸之影像超連結：此議題主要是考量人類的閱讀行為，通常學習者的閱讀習慣是由左至右且由上到下，故製作一份網頁教材盡可能不超

出瀏覽器的寬度，以免造成學習上的不便。以行動裝置瀏覽一份教材而言，目前大部分的瀏覽器皆提供『雙向捲軸』功能。當我們下載一份完整的靜態文件至行動裝置，瀏覽器會重新編排文字及影像，若影像超出行動裝置的螢幕大小，那麼則透過雙向捲軸來瀏覽，如圖 17所示。另一種情況則是當學習者認為原始影像有礙於閱讀，那麼作法上則會將原始網頁 *Page_{source}* 之影像按行動裝置之螢幕比例來縮小，並產生一份新的網頁 *Page_{new}*，兩份網頁之間建立起超連結關係以便學習者切換。在後面將討論到內文感知行動學習教材的建置，我們也是利用類似的觀念。

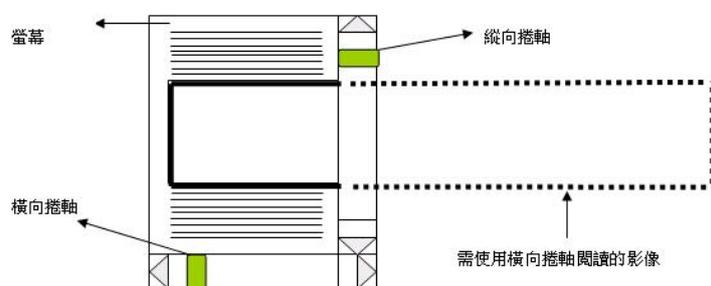


圖 17：雙向捲軸之瀏覽示意圖

以上對於學習者如何利用行動裝置來存取一份靜態文件作了完整的描述。為了達到最佳化的內容呈現，在內容最適化 (Content Adaptation) 的技術上主要分為靜態與動態兩種。以靜態最適化 (Static Adaptation) 而言，根據行動裝置的不同，系統為其準備不同的呈現樣式。例如當學習者利用螢幕大小為 200*200 的 PDA 來存取一份靜態文件，那麼系統便會挑選最適合的呈現樣式回應給學習者。這種作法的缺點就是缺乏彈性，因為若學習者所使用的行動裝置，系統本身並沒有提供的話，那麼就必須重新定義一份新的呈現樣式，這麼一來勢必造成儲存空間上的浪費。而動態最適化 (Dynamic Adaptation) 所採取的作法則是當學習者透過行動裝置來存取教材時，系統會自動去偵測行動裝置的設備內文，並立即產生符合的呈現樣式。在本研究中，我們採用動態最適化的技術，透過 Context Detector 來偵測設備內文，以達到最佳化的內容呈現 [36][37][38]。

3.4 行動學習教材解析

在前兩個小節，我們已經了解一份靜態文件的解析及存取過程，在 3.4 節與 3.5 節的重點則會放在一份行動學習教材的解析與建置。就本論文而言，行動學習教材是由靜態文件與動態文件整合而成。有別於傳統，一份行動學習教材除了靜態文件以外還增加了動態文件，而所謂的動態文件的組成元素主要為視訊檔 (Video Files) 與音訊檔 (Audio Files)。例如學習者透過行動裝置下載了一份行動學習教材，其中的內容包含一個投影片檔案 (powerpoint) 以供學習者瀏覽，另外還包含了一個視訊檔案 (video)，其內容是老師講解投影片的過程，以輔助學習者更了解這份教材。行動學習教材之解析主要是靜態文件之解析 (請參照圖 8) 的延伸，如圖 18 所示。

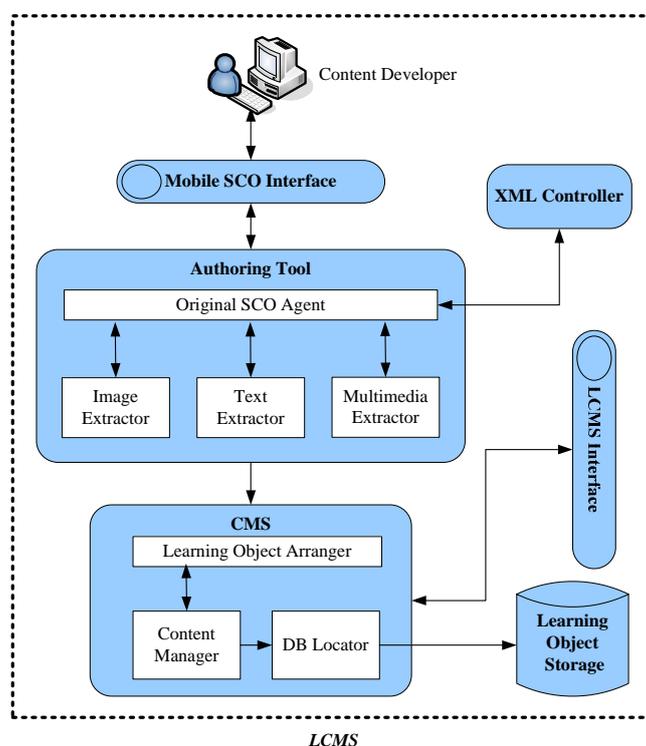


圖 18：行動學習教材解析架構圖

行動學習教材的解析架構基本上是延續著靜態文件的解析模式再加以擴

充。透過Original SCO Interface，教材編輯者可以上傳靜態文件和動態文件至Authoring Tool，再由Authoring Tool中的Original SCO Agent來解析上傳文件的架構及組成元素。如同靜態文件解析架構，Image Extractor與Text Extractor負責擷取影像與文字兩種元素並定義其元素層級，另外為了處理動態文件中的元素，我們則利用Multimedia Extractor來擷取視訊檔或音訊檔。當上傳的文件都解析完成，Original SCO Agent則會呼叫XML Controller建立一份XML文件（Manifest File）來描述各元素之間的架構關係，如圖 19所示。當XML文件建置完成，Authoring Tool中的Original SCO Agent便會將工作權交給CMS中的Content Manager，其過程則依循靜態文件的模式來處理，最後再呼叫DB Locator，將這些元素依序存放到資料庫。

```
<?xml version="1.0" standalone="no" ?>
- <manifest identifier="AAAA-BBBB-CCCC-DDDD-2005-0316-0000" xmlns="http://www.imsglobal.org/xsd/imscp_v1p1"
  xmlns:adlcp="http://www.adlnet.org/xsd/adlcp_v1p3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.imsglobal.org/xsd/imscp_v1p1 imscp_v1p1.xsd http://www.adlnet.org/xsd/adlcp_v1p3
  adlcp_v1p3.xsd">
  <metadata />
  <organizations />
- <resources>
  - <resource identifier="AAAA-BBBB-CCCC-DDDD-2005-0316-0000-R" type="webcontent" href="AAAA-BBBB-CCCC-DDDD-2005-
    0316-0000.html" adlcp:scormType="sco">
    <file href="media/OutputFile.wmv" />
    <file href="slides/1.gif" />
    <file href="slides/2.gif" />
    <file href="slides/3.gif" />
    <file href="slides/4.gif" />
    <file href="slides/5.gif" />
    <file href="slides/6.gif" />
    <file href="slides/7.gif" />
    <file href="slides/8.gif" />
    <file href="slides/9.gif" />
    <file href="slides/10.gif" />
    </resource>
  </resources>
</manifest>
```

圖 19：Manifest 之檔案範例（imsmanifest.xml）

假設一份行動學習教材是由投影片檔案（powerpoint）與多媒體檔案整合而成，其中多媒體的部分是授課者講解投影片的視訊檔（video）。以技術層面而言，Original SCO Agent 利用 Image Extractor 與 Text Extractor 把投影片的影像與文字擷取出來並轉換成 HTML 以解析其標籤，且利用 Multimedia Extractor 將視訊檔的內容依據每一頁投影片來切割，最後再建立投影片與視訊檔的關聯性，細部的處理上，可分為以下三個部分：

- 影像解析：Original SCO Agent利用Image Extractor來解析一份投影片中
 的影像。在本研究中，影像可分成兩個類型來探討，一種是『投影片每
 一頁之影像』，另一種則是『投影片內容之影像』。Image Extractor首先
 將每一頁投影片存成影像檔，其目的是為了之後建置行動學習教材時，
 能夠維持教材的完整性，就技術上而言則可直接透過Microsoft
 Powerpoint將一份投影片另存成影像檔，如圖 20所示。除此之外，Image
 Extractor還必須解析投影片中的個別影像，由於投影片的組成元素是各
 自獨立的元件，藉由捨棄影像以外的元件，剩下來的即為投影片內容之
 影像，如圖 21所示。

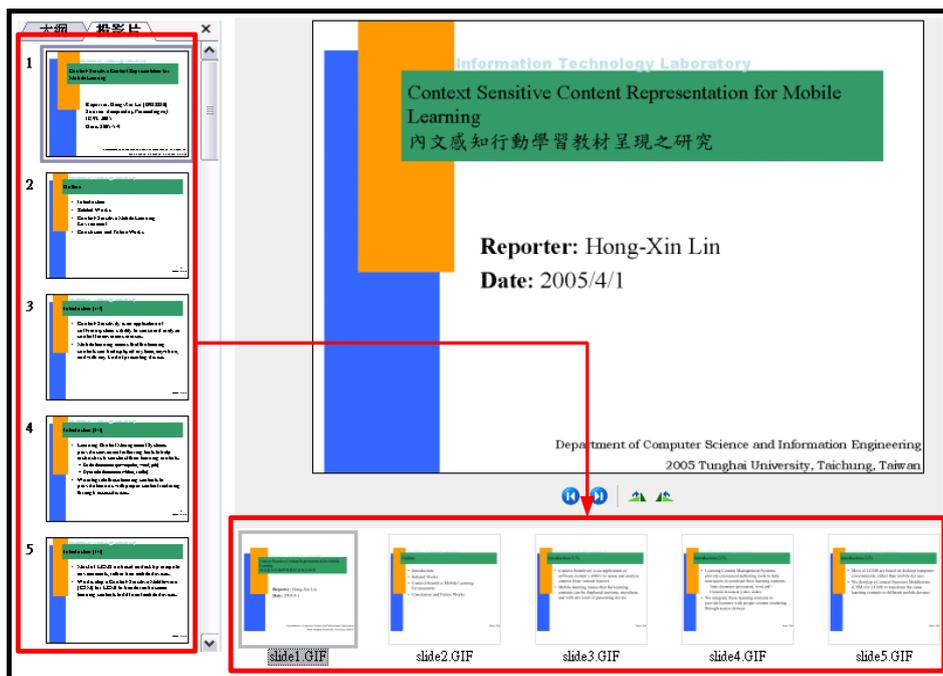


圖 20：每一頁投影片之影像解析示意圖

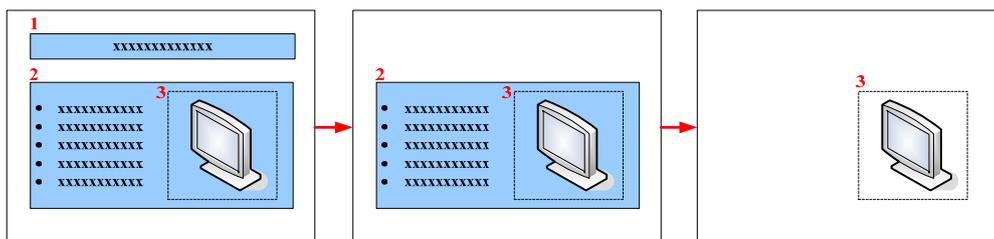


圖 21：投影片內容之影像解析示意圖

- 文字解析：Original SCO Agent利用Text Extractor來解析一份投影片的文字。一份投影片的文字可分為『大綱部分』與『註解部分』。大綱部分也就是每一張投影片的實際內容，可以透過大綱模式來檢視，如圖 22所示。技術上可直接透過Microsoft Powerpoint將一份投影片另存成大綱模式以取得此部分的文字。若只有解析大綱部分對於學習者而言是不夠的，Text Extractor還必須解析註解部分以讓學習者更了解投影片的內容，而註解部分當然是教材編輯者所提供的，如圖 23所示。程式撰寫上可利用Interop.Microsoft.Office.Core.dll與Interop.PowerPoint.dll兩支元件（Component）來達成目的。

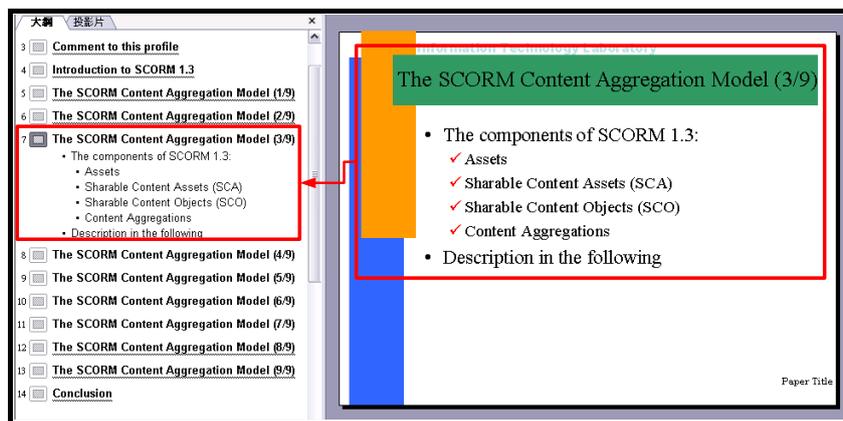


圖 22：投影片之大綱文字解析示意圖

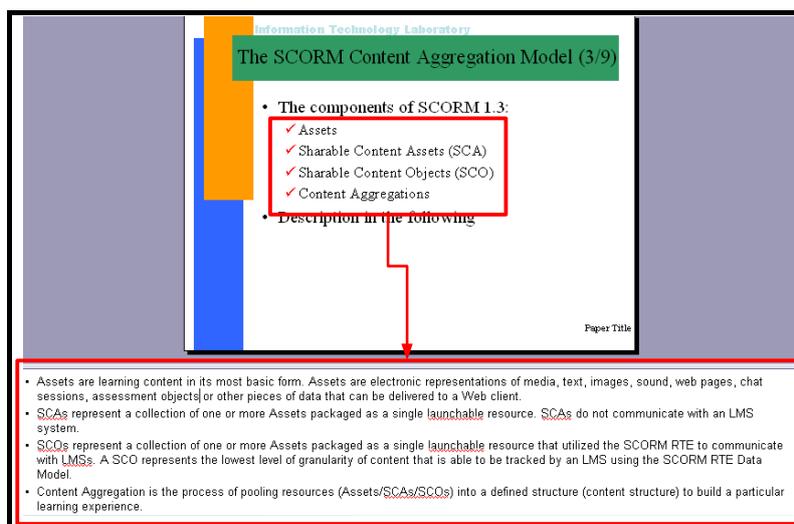


圖 23：投影片之註解文字解析示意圖

- 多媒體解析：當Image Extractor與Text Extractor解析完一份投影片的組成元素之後，Original SCO Agent便會呼叫Multimedia Extractor來解析動態文件，在這裡指的是授課過程的視訊檔或音訊檔。Multimedia Extractor會將視訊檔的內容依據每一頁投影片的講解過程透過Metadata來描述以示區隔，如圖 24所示。這種作法的好處是學習者可以快速且有效率地得到自己所要的資訊，例如學習者已瀏覽至第五頁的投影片且系統正播放此頁的視訊檔，而當學習者欲重新瀏覽之前的某一頁，那麼系統便可以快速地找到視訊檔之內容相對應的部分並且重新播放。

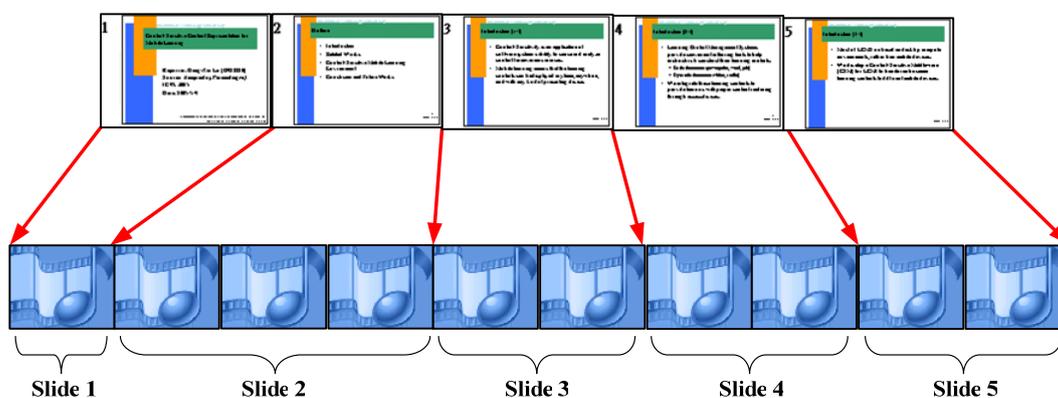


圖 24：多媒體解析示意圖

以上完整地說明了行動學習教材的解析過程，首先由 Authoring Tool 中的 Original SCO Agent 利用 Image Extractor、Text Extractor 與 Multimedia Extractor 來解析一份行動學習教材的組成元素，解析完畢再透過 CMS 來管理教材並歸類到資料庫。以下則會探討一份內文感知行動學習教材的建置過程。

3.5 內文感知行動學習教材建置

內文感知行動學習教材可定義為：當學習者欲透過不同的行動裝置來存取一份學習教材，則系統會偵測行動裝置的設備內文，透過動態最適化的技術建置一份行動學習教材，以提供給學習者最佳化的內容呈現。而內文感知行動學習教材

的建置則可分為：行動學習教材之解析與行動學習教材之存取。關於行動學習教材的解析，已在上一節中清楚地描述，故本小節的重點則會放在行動學習教材的存取。透過 *Learners and Learning Devices* 與 *Context-Sensitive Middleware* 之間的關聯性可用來說明一份內文感知行動學習教材的建置過程，如圖 25 所示。

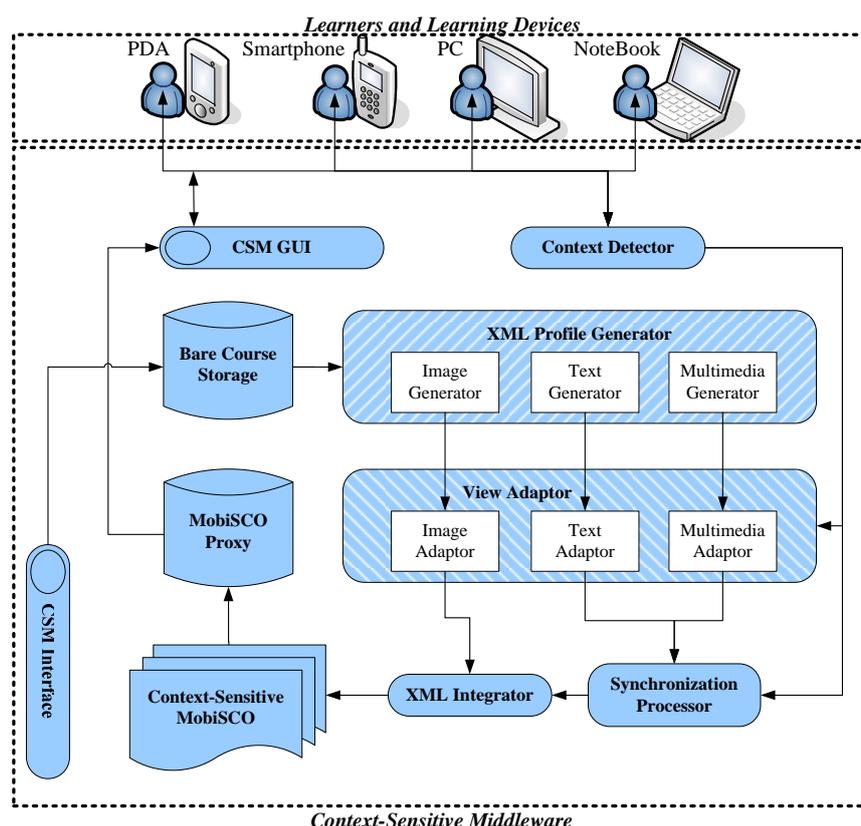


圖 25：行動學習教材建置架構圖

Context-Sensitive Middleware 主要是由有兩項外部事件所觸發：一項為透過 *CSM Interface* 來接受由 *LCMS* 所提供的原始教材並且將教材存放到 *Bare Course Storage*。另一項則藉由 *CSM GUI* 讓學習者透過不同的行動裝置來存取系統。在架構圖中的 *Learner and Learning Devices* 說明了學習者可以利用 *PDA*、*Smartphone*、*PC* 或 *Notebook* 來存取系統。假設學習者選擇了 *PDA* 作為學習工具，那麼連結到系統的同時 *Context Detector* 便開始偵測其設備內文，當學習者透過 *CSM GUI* 選擇了一份教材，那麼系統便會先在 *MobiSCO Proxy* 中搜尋是否有符合此 *PDA* 的樣式 (*Stylesheet*)，如果存在的話則一份行動學習教材可立即建

置完成。倘若 MobiSCO Proxy 不存在符合的樣式，那麼系統會從 Bare Course Storage 取出原始教材資源 (Image、Text 與 Multimedia)，先利用 XML Profile Generator 來建立描述這些資源的 XML 文件，再利用 View Adaptor 與 Context Detector 作最適化的處理並且產生一份新的樣式。然後利用 Synchronization Processor 與 XML Integrator 建立教材資源彼此之間的連結關係並且產生 HTML，最後整合成一份 Context-Sensitive MobiSCO。上述大致是一份內文感知行動學習教材的建置過程，以下則針對系統中兩個重要的模組作細部的說明：

➤ XML Profile Generator

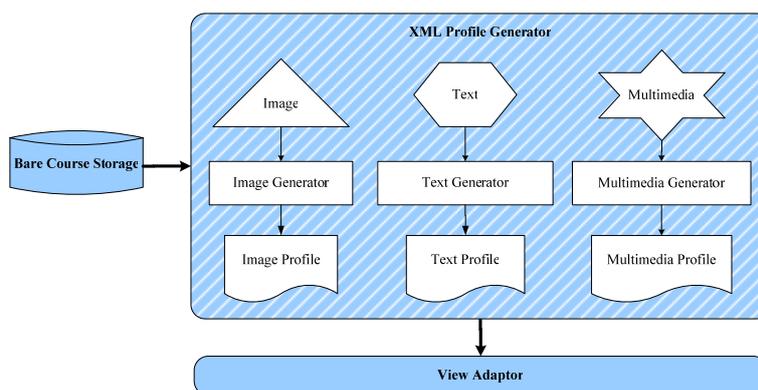


圖 26：XML Profile Generator 架構圖

由於行動裝置所提供的資源有限，以小面積之螢幕為例，一份powerpoint之投影片主要是針對PC環境所設計而無法完整地呈現在行動裝置上，故必須藉由分析與轉換的技術來克服此問題。為了讓學習者利用行動裝置來瀏覽一份教材，所有的教材資源首先必須透過XML Profile Generator來建立描述這些資源的XML文件，如圖 26所示。

XML Profile Generator 把教材資源分成三類：Image、Text 與 Multimedia。Image 是指一份投影片的影像元素，包含『投影片每一頁之影像』與『投影片內容之影像』。Text 是指一份投影片的文字元素，包含『大綱部分』與『註解部分』。

Multimedia 則是指一份動態文件，包含視訊檔或音訊檔。這三種資源必須透過 XML Profile Generator 轉換成相對應的 XML 文件。

在前一節已討論到投影片中的文字可解析成純文字檔，那麼就可以針對此檔案來建立描述Text的XML文件，就技術上而言則是藉由文字元素中的項目符號來判斷XML的標籤，例如『•』為投影片中的主項目，XML之標籤為<CourseList />，『-』為投影片中的次項目，XML之標籤為<List />，如圖 27所示。除此之外，建立Image或Multimedia的XML文件則是藉由描述這些資源的檔案路徑，XML 之標籤為<File />，如圖 28所示。當全部的XML文件都建立完畢，View Adaptor便可以將這些文件，依據不同的行動裝置作最適化的處理。

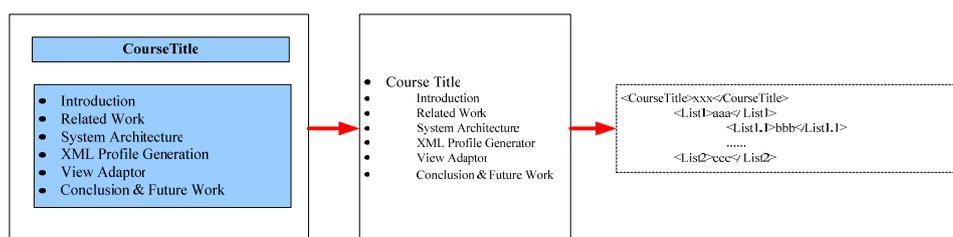


圖 27：建立 XML 文件示意圖之一

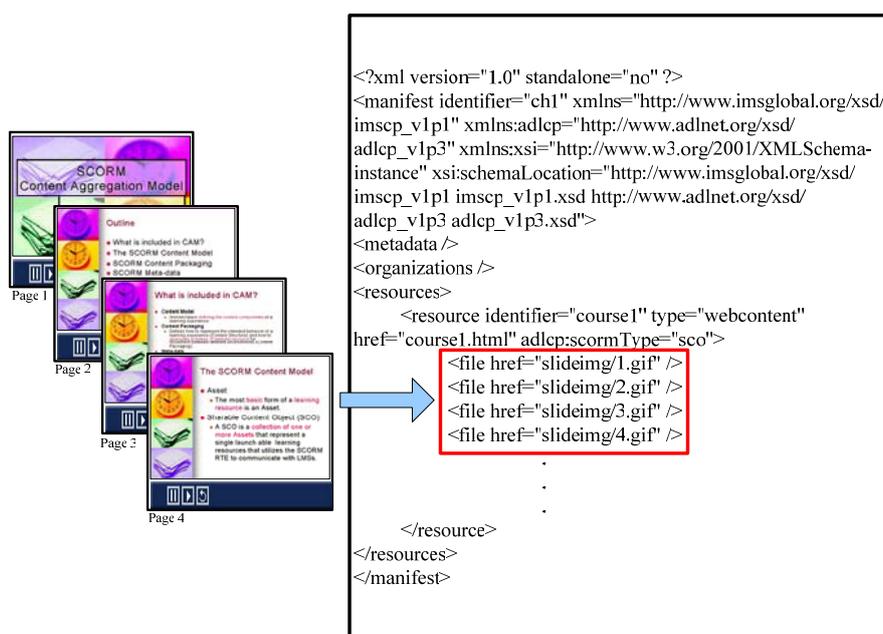


圖 28：建立 XML 文件示意圖之二

➤ View Adaptor

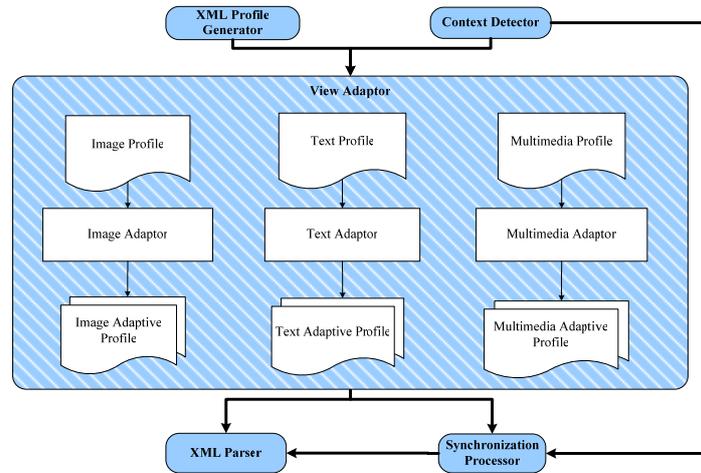


圖 29：View Adaptor 架構圖

表 3：行動設備內文

1.	處理機效能
2.	記憶體大小
3.	是否支援 Gray-level Image
4.	是否支援 High Pixel Color Image
5.	螢幕大小
6.	螢幕解析度
7.	是否支援雙捲軸
8.	儲存體大小
9.	是否支援 HTML
10.	是否支援 XHTML
11.	是否支援視訊檔播放
12.	是否支援音訊檔播放

XML Profile Generator已建立了描述教材資源的XML文件，在這裡是指Image Profile、Text Profile與Multimedia Profile，然後交給View Adaptor處理。View Adaptor主要的工作就是選擇適合的樣式 (Stylesheet) 套用在這些XML文件上，根據Context Detector偵測到的設備內文來決定，如圖 29所示。所謂的內文是指軟體本身、終端設備、軟體使用者、終端設備環境等可以被偵測的相關事物。本研究中的設備內文則是探討各種行動裝置本身的環境限制。所謂的內文感知

(Context Sensitivity) 是指系統本身有能力偵測並且分析來自不同的設備內文。為了建置一份內文感知行動學習教材，我們需考量各種設備內文所帶來的影響，如表 3 所示。為了能夠順利處理表 3 的所有限制，透過 R. Mohon 等人所提出的基本方法來解決，由於目前的行動裝置提供了更豐富的資源，欲讓這些新機種達到最佳的呈現環境，許多客制化的機制也必須被考量到 [36][37][38]。

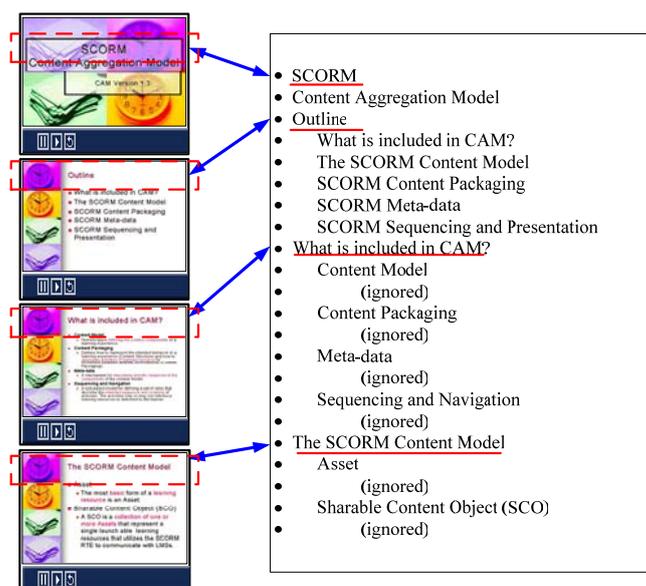


圖 30：行動學習教材資源之連結關係

原始的教材資源經過 XML Profile Generator 與 View Adaptor 的處理之後，一份內文感知行動學習教材已初步建置完成。接下來所必須做的則是教材之間的同步與整合，透過 Synchronization Processor 與 XML Integrator 來完成。技術上而言首先是先建立 Image 與 Text 之間的雙向連結關係，再建立 Image 與 Multimedia 的連結關係，最後則產生 HTML 以呈現在行動裝置的瀏覽器上，如圖 30 所示。當學習者得到一份行動學習教材，可先從 Image 了解到一份教材的原貌並且根據自己的意願選擇是否播放 Multimedia，如果想知道更深入的資訊則透過 Text 來了解。如此一份內文感知行動學習教材已建置完成，我們稱為一份 Context-Sensitive MobiSCO。

第 4 章 研究成果

本研究主要利用 MVC Pattern 來規範整個行動學習環境的架構，以達到系統容易維護與提升元件的可再用性。以 *Model* 的角度而言，為了提升學習教材的可再用性並且與國際標準接軌，我們採用 SCORM 1.3.1 將學習教材分為 Asset、SCA、SCO 與 Content Packaging，利用 XML 來描述這些元素，並且建立元素之間的連結關係。以 *View* 的角度而言，Context-Sensitive Middleware 針對各種行動裝置作最適化的內容呈現。至於 *Controller* 的工作則是處理內文感知的分析與偵測，並且根據行動設備內文產生呈現的方法與策略，三者間的互動關係如圖 31 所示。當 Controller 接收到 User Interaction，依據偵測與分析的 Context 來決定呈現模式，並且通知 View 依照此模式作內容呈現。而 View 則會向 Model 提出呈現時所需要的內容。當呈現的內容需要作變更時，Controller 會發出訊息通知 Model 以進行資料變動，然後再通知 View 作相對應的改變。

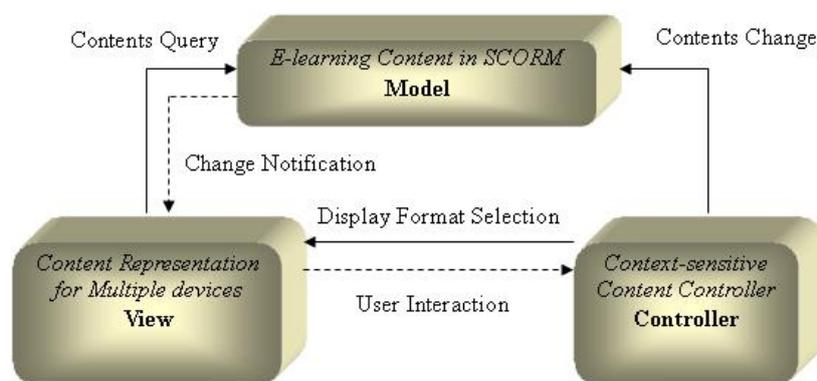


圖 31：MVC Pattern 互動關係示意圖

4.1 實驗平台

在本研究中，系統開發軟體為 Microsoft Visual Studio .NET 2003，程式語言採用 VC#.NET。實驗平台之伺服器端 (Server) 主要為 Pentium(R) 4 CPU 2.40GHz 與 512MB RAM 的桌上型 PC，作業系統為 Windows XP。用戶端 (Client) 則為

Microsoft Visual Studio .NET 2003 之 Pocket PC 2002 模擬器、Mio 8390 Smartphone 與 HP iPAQ h6300 系列的 Pocket PC。

4.2 實驗結果

為了建立內文感知行動學習環境，我們的實驗結果主要分為兩個部分：首先是運用第三章的理論來實作 Context-Sensitive Middleware，其目的在於將原始教材轉換成一份 MobiSCO。另外一項實驗則是把建置完成的 MobiSCO 下載至用戶端作最適化的內容呈現。

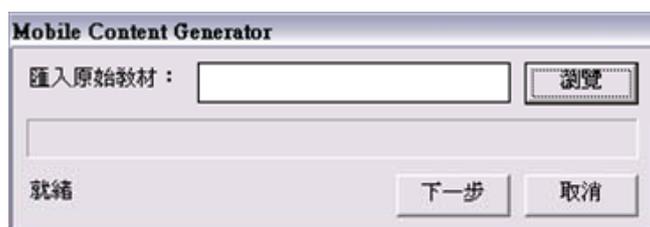


圖 32：Context-Sensitive Middleware 之操作介面

Context-Sensitive Middleware 的設計考量在於使用簡單且操作方便，如圖 32 所示。欲建置一份 MobiSCO，首先點選系統中『瀏覽』以選擇原始教材之壓縮檔 (zip file)，如圖 33 所示。



圖 33：行動學習教材建置之一

當選擇完一份原始教材，這份教材會被上傳到系統資料夾（預設路徑為 C:\ICAN\SYSTEMp）並且解壓縮成三種教材資源，如圖 34所示。點選『下一步』，系統隨即列出三種教材資源的路徑，然後點選『建置』，如圖 35所示。



圖 34：行動學習教材建置之二



圖 35：行動學習教材建置之三

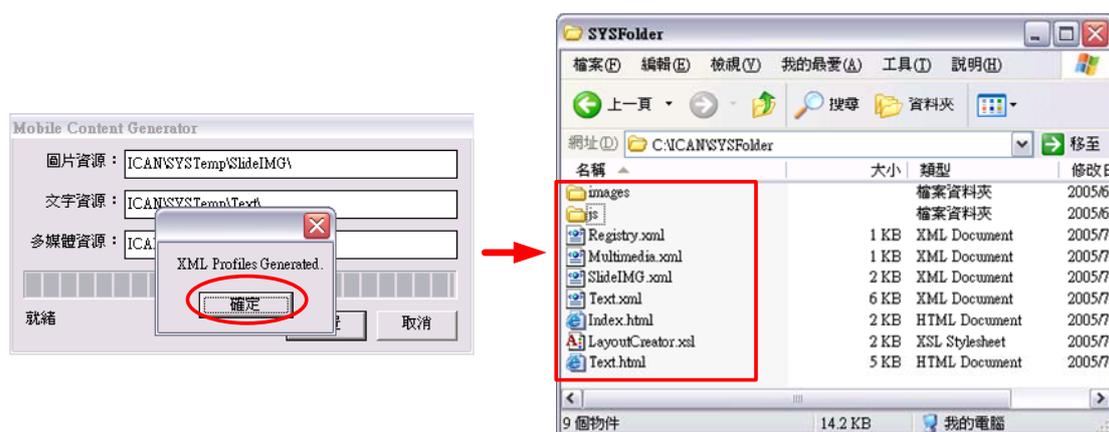


圖 36：行動學習教材建置之四

如圖 36所示，系統會自動產生三份XML文件，分別是SlideIMG.xml、Text.xml

與Multimedia.xml以描述三種教材資源，然後跳出一個小視窗來告知使用者，點選『確定』之後，系統會依據各種行動裝置來產生動態最適化的XSL檔（LayoutCreator.xsl）與HTML檔（Index.html、Text.html）。以下是三份XML文件與XSL文件之範例檔案，如圖 37、圖 38所示。

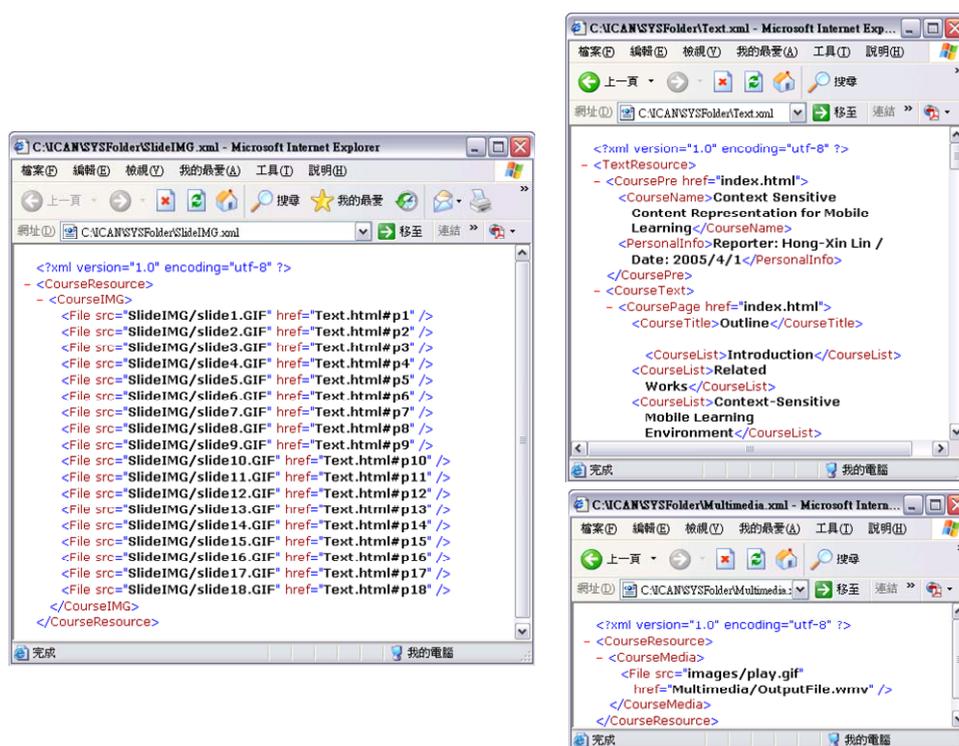


圖 37：教材資源之 XML 文件範例

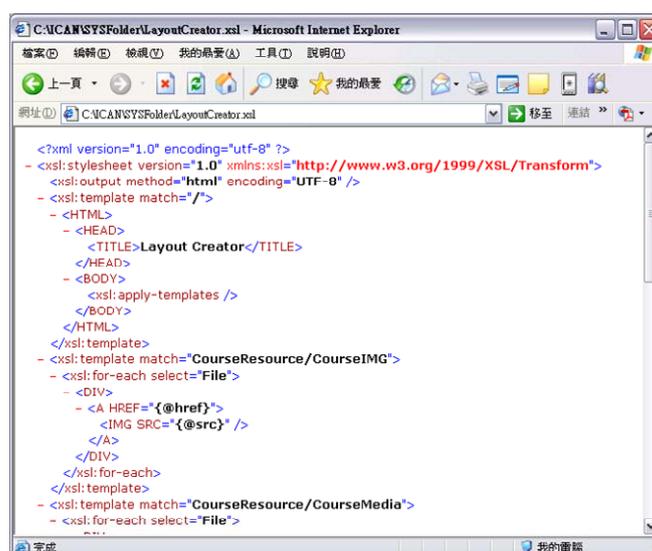


圖 38：教材資源之 XSL 文件範例

我們將一份MobiSCO呈現在一般的桌上型PC，如圖 39、圖 40所示。學習者利用IE瀏覽器開啟Index.html，其內容主要是用來呈現教材的Image與Multimedia，學習者可以自行播放授課者的視訊檔。當學習者點選到某一張Image，則立即連結到Text.html，其內容是用來呈現教材的Text。Index.html與Text.html彼此建立了雙向超連結以便學習者可以隨時切換。

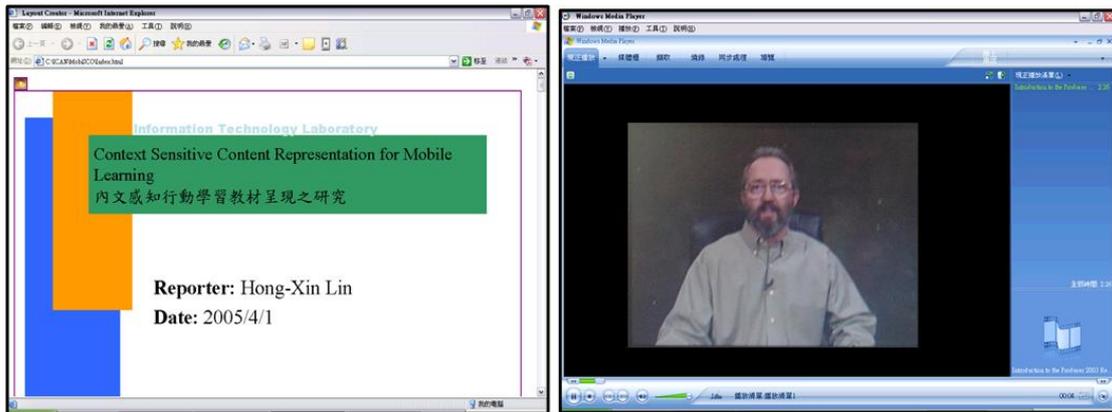


圖 39：行動學習教材之 HTML 範例檔 (Index.html)

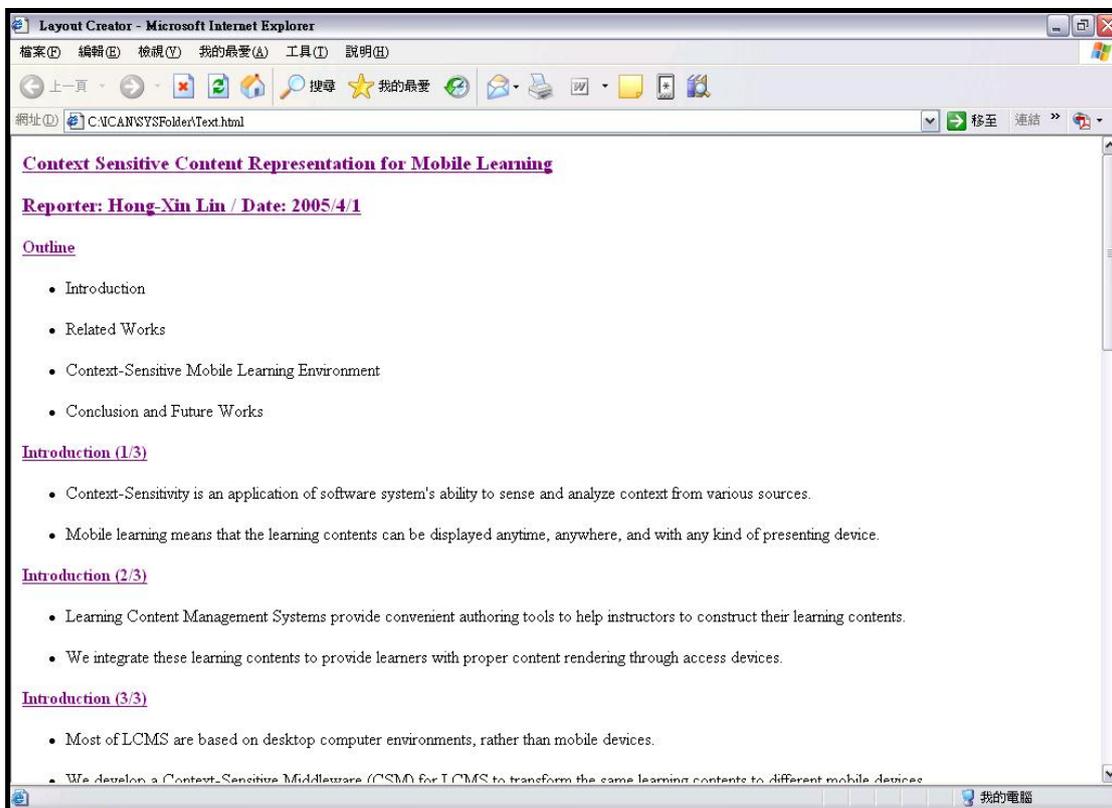


圖 40：行動學習教材之 HTML 範例檔 (Text.html)

以下是一份MobiSCO呈现在Mio 8390之范例，如图 41所示。



圖 41：MobiSCO 於 Mio 8390 之呈現

以下是一份MobiSCO呈现在HP iPAQ h6300之范例，如图 42所示。



圖 42：MobiSCO 於 HP iPAQ h6300 之呈現

第 5 章 結論與未來方向

在普及運算與無限網路環境日趨成熟的情況下，本研究致力於打造一個全方位的內文感知行動學習環境，讓學習者能夠在任何時間、任何地點以及任何種類的設備下學習知識並且得到收穫，所以透過行動裝置來達到學習目的是可行且必要的。在研究的過程當中，我們發現了一些值得深入研究的議題，例如行動裝置的異質性使得內容呈現必須客制化，動態最適化的技術來偵測設備內文，行動學習教材之內容範疇等議題。

本論文提出一個內文感知的中介軟體，透過此中介軟體，可將一份原本只能呈現在桌上型裝置的教材，依據行動裝置的不同作最適化的呈現，以達到行動學習的目的。論文中首先探討靜態文件的解析過程，一份靜態文件之組成元素為影像與文字，系統利用 Image Extractor 與 Text Extractor 將這些元素擷取出來並建立符合 SCORM 的階層架構。此外則針對靜態文件之版面佈置、影像解析與文字解析作更細部的描述。接下來則是靜態文件的存取過程，學習者可透過不同的行動裝置來存取一份靜態文件且系統利用 Context Detector 來偵測設備內文。另外則透過網頁切割與網頁超連結、調整影像大小乃至於影像超連結來達到內容呈現的最佳化。而行動學習教材與之前所提最大不同之處在於新增了動態多媒體資源，故需透過 Multimedia Extractor 來解析此元素。當學習者欲透過不同的行動裝置來存取一份學習教材，則系統會偵測行動裝置的內文，透過動態最適化的技術建置一份行動學習教材，以提供學習者最佳化的內容呈現。我們將內文感知行動學習教材稱為 Context-Sensitive MobiSCO。

本研究將重點放在一份內文感知行動學習教材之內容呈現，故考量的議題主要是行動裝置的設備內文偵測與學習教材呈現的關係，因此尚有一些不足的部分可待加強或擴充，分述如下：

1. 學習者學習特徵之萃取：一份完整且適合學習者的教材除了最佳化的內容呈現外，還必須考量不同學習者的學習特徵（Learning Characteristic）以提供多樣性的教材內容，透過學習特徵資料的分析來豐富教材內容，使之更能符合學習者的需要，另外也可以在系統中增設學習者學習項目的偏好設定，讓他們可以根據各別的需求來做彈性的選擇。
2. 線上評量考試：考試的目的一方面可了解學習者的學習效果，另一方面則可作為教材的修正方針，透過系統自動化建置一份評量考試題目是我們的目標。同一份線上評量考試是否符合每一位學習者，或者是要根據吸收程度作難易度的調整，這些都是值得深入研究與探討的議題。
3. 師生互動式教學：在系統中增設學習論壇，讓師生在互動過程中，老師可以知道學生吸收了多少而學生則可隨時隨地提問題。另外行動裝置的可攜性有助於戶外教學的應用，若能與博物館或科博館合作以獲得相關的資訊勢必可以更落實行動學習。

在未來，我們將針對『內容呈現』、『學習者學習特徵』、『線上評量考試』、『互動式學習』乃至於『離線學習』等議題作更深入的研究進而彌補上述的不足之處，以期能夠實現全方位的行動學習環境並解決數位學習領域所會面臨到的各種問題，這些都是在未來需要更努力的目標。

參考文獻

- [1] D. Garlan, D. Siewiorek, A. Smailagic and P. Steenkiste, “Integrated Pervasive Computing Environments,” *IEEE Pervasive Computing*, Vol. 21, No. 2, 2002, pp. 22-31.
- [2] A. C. Ikeji, F. Fotouhi, “An adaptive real-time Web search engine,” *Proceedings of the second international workshop on on Web information and data management*, 1999, pp. 2-6.
- [3] L. Huang, M. Hemmje and E.J. Neuhold, ” ADMIRE:an adaptive data model for meta search engine,” *Computer Network* 33, 2000, pp. 431-448.
- [4] 高台茜, [未來教室學習 - 以無線網路應用為基礎的認知學徒制學習環境], 台大教與學網站, 九十一年十二月十日。
- [5] A. Dix, T. Rodden, N. Davies, Jonathan Trevor, Adrian Friday, and Kevin Palfreyman, “Exploiting space and location as a design framework for interactive mobile systems,” *ACM Transactions on Computer-Human Interaction*, September 2000.
- [6] IMS Metadata Specification, Available to <http://xml.coverpages.org/ims.html>
- [7] IEEE Learning Technology Standards Committee (LTSC), Available to <http://ltsc.ieee.org/>
- [8] Aviation Industry Computer-based Training Committee, Available to <http://elearning.ksut.edu.tw/Resource/epaper/20021105/02.html>
- [9] Advanced Distributed Learning (ADL), “SCORM 2004 2nd Edition Overview,” Available at *ADLNet.org*, July, 2004.
- [10] Advanced Distributed Learning (ADL), “SCORM Content Aggregation Model Version 1.3.1,” Available at *ADLNet.org*, July, 2004.
- [11] Advanced Distributed Learning (ADL), “SCORM Run-Time Environment Version 1.3.1,” Available at *ADLNet.org*, July, 2004.
- [12] Advanced Distributed Learning (ADL), “SCORM Sequencing and Navigation Version 1.3.1,” Available at *ADLNet.org*, July, 2004.
- [13] Learning Material Markup Language (LMML), Available to <http://xml.coverpages.org/lmml.html>
- [14] F.Buendía, J.V Benlloch, J. M. Gomez, J.C. Burguillo, D. A. Rodriguez, “Introducing LMML into the development of didactic resources on IEE,” Vol. 1, 2002.
- [15] Christian Süß, Burkhard Freitag, “Learning Material Markup Language LMML,” IFIS-Report, 03, 2001.
- [16] Alliance of Remote Instructional Authoring and Distributed Networks for Europe, Available to <http://elearning.ksut.edu.tw/Resource/epaper/20021105/02.html>

- [17] Oliver bohl, Jorg Schellhase, Ruth Sengler and Udo Winand, "The Sharable Content Object Reference Model (SCORM) – A Critical View," *Proceedings of the International Conference on Computers in Educations (ICCE' 02)*, 2002.
- [18] Jin-Tan David Yang and Chun-Yen Tsai, "An Implementation of SCORM-compliant Learning Content Management System – Content Repository Management System," *Proceedings of the 3rd IEEE International Conference on Advanced Learning Technologies (ICALT' 03)*, 2003.
- [19] Timothy K. Shih, Wen-Chih Chang, Nigel H. Lin, Louis H. Lin, Hun-Hui Hsu and Ching-Tang Hsieh, "Using SOAP and .NET Web Service to Build SCORM RTE and LMC," *Proceedings of the 17th International Conference on Advanced Information Networking and Applications (AINA' 03)*, 2003.
- [20] Peiya Liu, Liang H. Hsu and Amit Chakraborty, "Towards Automating the Generation of SCORM-Based Multimedia Product Training Manuals," *Proceedings of 2002 IEEE International Conference Multimedia and Expo (ICME' 02)*, 2002.
- [21] Nigel H. Lin, Timothy K. Shih, Hui-huang Hsu, Hsuan-Pu Chang, Han-Bin Chang, Wen Chieh Ko and L. J. Lin "Pocket SCORM," *Proceedings of the 24th International Conference on Distributed Computing Systems Workshops (ICDCSW'04)*, March, 2004, pp. 274-279.
- [22] Learning Management System (LMS), Available to <http://www.elearningpost.com/features/archives/001022.asp>
- [23] M. Brennan, S. Funke, C. Anderson, "The Learning Content Management System - A New eLearning Market Segment Emerges," *An IDC White Paper*, December, 2001.
- [24] Dey, A. K., Abowd, G. D., "Toward a Better Understanding of Context andContext-awareness," *Georgia Tech GVU Technical Report*, GIT-GVU-99-22, 1999.
- [25] Schilit, B., Theimer, M., "Disseminating Active Map Information to MobileHosts," *Proceedings of IEEE Network*, vol. 8, issue 5, 1994, pp.22-32.
- [26] Ryan, N., Pascoe, J., Morse, D., "Enhanced Reality Fieldwork: the Context-aware Archaeological Assistant," Gaffney, V., van Leusen, M., Exxon, S., (eds.)*Computer Applications in Archaeology*, 1997.
- [27] Franklin, D., Flaschbart, J., "All Gadget and No Representation Makes Jack a Dull Environment," AAI 1998 Spring Symposium on Intelligent Environments, *Technical Report SS-98-02*, 1998, pp. 155-160.
- [28] Ward, A., Jones, A., Hopper, A., "A New Location Technique for the Active Office," *Proceedings of IEEE Personal Communications*, vol. 4, issue 5, 1997, pp.42-47.
- [29] Rodden, T., Cheverst, K., Davies, K., Dix, A., "Exploiting Context in HCI

Design for Mobile Systems,” *Proceedings of Workshop on Human Computer Interaction with Mobile Devices*, 1998.

[30] Dey, A. K., Abowd, G. D., “Toward a Better Understanding of Context and Context-awareness,” *Georgia Tech GVU Technical Report*, GIT-GVU-99-22, 1999.

[31] Chen, G., Kotz, D., “A Survey of Context-aware Mobile Computing Research,” *Dartmouth Computer Science Technical Report TR2000-381*.

[32] Alvin T. S. Chan and Siu-Nam Chuang, “MobiPADS: A Reflective Middleware for Context-Aware Mobile Computing,” *IEEE Transaction on Software Engineering*, VOL. 29, No.12, December, 2003.

[33] 韓世翔，[考量環境智慧之適性化行動學習平台]，中原大學資訊管理學系碩士論文，九十三年八月。

[34] W3C, Available to <http://www.w3c.org>

[35] T. Reenskaug, “Working With Objects: The OOram Software Engineering Method,” Manning Publications, 0-13-452930-8, Per World and Odd Arild Lehne, 1996.s

[36] William C. Chu, Y. W. Chen, J. N. Chen, “Context-sensitive Content Representation for Static Document,” *Proceedings of 2004 Symposium on E-Learning*, 2004, pp. 111-123.

[37] R. Mohan, J. R. Smith, C. S. Li, “Adapting Multimedia Internet Content for Universal Access,” *IEEE Transaction on Multimedia*, Vol. 1, No. 1, 1999, pp. 104-114.

[38] 陳翼汶，[內文感知模式與靜態文件內容呈現之研究]，東海大學資訊工程與科學所碩士論文，九十三年六月。